

Nr. 6/86 Juni -DM 5,50 / ÖS 46 / SRF 5,50

TI-FORTH: EINE MÄCHTIGE SPRACHE

MODUL-EXPANDER NAVARONE

JOYSTICK OHNE ADAPTER

IMPRESSUM

TI-REVUE, die Zeitschrift für den TI PC und TI 99/4A erscheint monatlich in der AKTUELL-GRUPPE, Elisabethstraße 1, 8044 Lohhof.
Redaktion: Senator-Presse-Service.
Verantwortlich für den Inhalt: Heiner Martin.
Verantwortlich für Anzeigen: Bruno Redase.
Verantwortlich für Listings: Hartmut König.
Alle: Postfach 1107, 8044 Lohhof. Anfragen bitte nur schriftlich.

Druck: Maier und Söhne
Es gilt die Honorarliste des Verlages. Für unaufgefordert eingesandte Manuskripte und Listings keine Haftung.
Bei Einsendung von Texten, Fotos und Programmträgern erteilt der Autor dem Verlag die Genehmigung für einen einmaligen Abdruck sowie die Aufnahme in den Programm-Service nach den Verlags-Sätzen! Alle in dieser Zeitschrift veröffentlichten Beiträge sind urheberrechtlich geschützt. Jedwede Verwertung ist untersagt. Nachdruck nur mit ausdrücklicher schriftlicher Zustimmung des Verlages. Namentlich gezeichnete Artikel geben nicht unbedingt die Meinung der Redaktion wieder.
Kein Anspruch auf Lieferung bei Ausfall durch höhere Gewalt.
Gerichtsstand: München
Geschäftsführer: Werner E. Seibt
Abo- und Kassetten-Service: Henny Rose Seibt

INHALT

Nr. 6/86/Juni

Grüß Gott Guten Tag

In dieser Ausgabe wollen wir uns wieder einmal stark mit dem „Eingemachten" am TI 99/4A beschäftigen. Dabei handelt es sich weniger um Assembler-Programme, sondern mehr um Erklärungen, wie was und warum es gemacht werden kann. Viele Anfragen erhielten wir auch nach dem Aufbau der Sektoren auf den Disketten. Auch dazu findet Ihr Erklärungen in diesem Heft. Etwas in Verzug geraten sind wir mit der versprochenen Bauanleitung für eine Centronics-Schnittstelle. Wir hatten ja noch im letzten Jahr einen engagierten Leser gesucht, der uns die Platine entwickelte. Aus den ganzen Angeboten wählten wir damals jemanden, der einen zuverlässigen Eindruck machte. Leider hat er diese Erwartungen enttäuscht, denn außer Vertröstungen haben wir bisher nichts mehr gehört. Es sieht so aus, daß wir hier nochmal von vorne beginnen müssen, so daß sich die Veröffentlichung noch weiter verzögert. Aus den USA kommen immer wieder Meldungen über interessante Software-Neuentwicklungen. Wir versuchen dabei natürlich immer am Ball zu bleiben, aber solange diese Programme von keinem deutschen Händler importiert werden, sieht das mit der kontinuierlichen Versorgung aller User hier nicht gut aus. Leider scheinen einige Händler etwas Angst vor neuen Investitionen in die TI-Produktpalette zu haben. Wir können das eigentlich nicht ganz verstehen, denn gerade in letzter Zeit haben ein paar Beispiele gezeigt, daß für wirkliche Neuheiten durchaus ein Markt vorhanden ist, nur: Uralte Module braucht heute wirklich fast (?) niemand mehr.
Leider ist es nun schon üblich geworden, an dieser Stelle über Gerüchte zu sprechen. Hier also das neueste in Bezug auf den neuen Computer aus den USA: Er soll jetzt als Karte für die Peripherie-Box auf den Markt kommen. An diese Karte kommt dann eine ganz normale IBM-Tastatur. Ein Anschluß für den Monitor und der Disk-Controller sei ebenfalls auf dieser Karte vorhanden.
Aber wie gesagt, gesehen hat auch diese Lösung bis jetzt noch niemand.
Bis zum nächsten Monat wünschen wir Euch viel Spaß mit dem TI 99/4A
Ihr TI-REVUE-Team

Nicht vergessen: Seid Ihr mit dem Heft zufrieden, sagt es weiter, seid Ihr unzufrieden, sagt es uns.
Und: Jeden Dienstag von 15 bis 19 Uhr stehen Ihnen unsere Lesertelefone zur Verfügung. Für technische und Assembler-Fragen Tel.-Nummer 0731/33220 und zu den Listings bzw. Fragen zu den Abonnements und dem Kassettenservice Tel.-Nummer 089/1298013.
Das Telefon für technische Fragen ist wegen Urlaubs des zuständigen Redakteurs am 10.6, 17.6. und am 24.6. nicht besetzt. Wir bitten um Verständnis dafür.

DIALOG

ZEICHENCODES 177 BIS 185 DEFINIEREN?

Wie kann ich die Zeichencodes 177 bis 185 definieren? Ist eine Lösung in Assembler denkbar? Und wie kann ich meinen Drucker dazu bewegen, nur 28 Zeichen breit zu drucken? Wie bewältigen Sie dieses Problem beim Ausdruck Ihrer Listings in der TI-REVUE? Oder benutzen Sie einen 28-Zeichen-Printer?
Peter Kliem,
Erftstadt

Unter der Annahme, daß sich die Frage auf den Betrieb in Basic oder Extended Basic bezieht, können wir dazu leider nur antworten: Die Zeichen 177 bis 185 können nicht definiert werden. Um Platz im VDP-RAM zu sparen, werden in Basic ja alle Zeichen mit einem Offset von 96 auf dem Bildschirm dargestellt. Der VDP kann aber nur 255 Zeichen darstellen, also ergäbe 177+ 96 mehr als 255 (genau 273). Wir benutzen zum Ausdruck ein besonderes Basic-Programm, d.h. zuerst wird das Listing mit LIST"DSK1.TEST" auf die Diskette gelistet. Hier ist es nun mit 80 Zeichen je Zeile vorhanden. Dies wird nun über ein Basic-Programm zum Ausdrucken geändert. Dieses Programm ist, soweit uns bekannt, bei der Fa. Hegasoft in München erhältlich. Mit einigen Druckern läßt sich aber auch ein anderes Verfahren anwenden: Diese Drucker müssen eine Einstellung des rechten und linken Randes besitzen, dann kann man diese auf z.B. linker Rand 10 und rechter Rand 38 einstellen. Wird nun ein LIST"RS232" durchgeführt, werden die Zeilen auf dem Papier nur 28 Zeichen breit geschrieben. Allerdings gibt es nach jeweils 80 Zeichen ein unschönes Weiterrücken in die nächste Zeile. Eine komplette Kopie der Bildschirmdarstellung wird damit also nicht erreicht.

FORMELSAMMLUNG, WAS IST DAS?

1. Beim TI-Modul "Schachmeister" erscheint im Hauptmenü an zweiter Stelle der Punkt "Formelsammlung". Wählt man jedoch Punkt 2 des Menüs an, gelangt man in das Programm "Schachmeister". Was hat es mit dem Punkt "Formelsammlung", der nicht angewählt werden kann, auf sich?
2. Gibt es außer dem TI-Modul noch weitere Schachprogramme, die „stärker" sind?

Stefan Lauterbach,
Goslar

Die "Formelsammlung" ist ein Relikt aus alten Zeiten des TI 99/4 (nicht 99/4A). Damals gab es wohl irgendwo eine eingebaute Software mit der Bezeichnung "Formelrechner". Nun baut ja das Schachmodul einen eigenen Bildschirm auf (zu sehen an den deutschen Meldungen im Titelbildschirm), d.h., es wird nicht die normale Routine nach dem Einschalten des TI 99/4A benutzt. Hier scheint nun im Schachmodul bzw. auch in einigen anderen Modulen, bei denen der Formelrechner im Hauptmenü erscheint, ein Fehler zu sein. "Stärkere" Schachprogramme für den TI 99/4A sind uns leider nicht bekannt.

PROGRAMMSPEICHERUNG IN STATISCHER SPEICHERERWEITERUNG

Ich habe eine Frage zu der Speichererweiterung aus Heft 9/85. Dank dem Tip der Redaktion (Invertieren) des DBIN-Signals brachte ich sie zum Laufen, ohne diese Änderung brachte die Speichererweiterung den Adressenhaushalt des TI etwas durcheinander.
Jetzt zu meiner Frage: Ich kann meine Daten in der Erweiterung nicht halten, obwohl ich den erwähnten Befehl benutzt habe (CALL LOAD, CALL PEEK). Auch die Spannung vom Akku ist hoch genug. Vielleicht können Sie mir helfen, wie nach dem Abschalten des TIs die Daten erhalten bleiben?
V. Brose,
Unna

Daten können in der Speichererweiterung nicht gehalten werden, jedoch Programme. Hier war unser Text in der TI-REVUE 9/85 wohl etwas unklar, denn wir erhielten ziemlich viele Anfragen. Deshalb hier nochmal die genaue Erklärung: Wenn ein Programm in der Erweiterung ist, wird vor dem Abschalten folgendes durchgeführt:
CALL PEEK (–31952, A1,A2,B1,B2)
PRINT A1,A2,B1,B2
Nun werden die auf dem Bildschirm stehenden Werte für A1, A2, B1 und B2 notiert (z.B. 254, 60,254,80). Nach dem Wiedereinschalten wird dann ein CALL LOAD (–31952,A1,A2,B1,B2, " ",–25,0) durchgeführt. Dabei sind für die Variablen A1, A2, B1 und B2 die vorher notierten Werte einzusetzen, also z.B.: CALL LOAD (–31952, 254,60,254,80, " ",–25,0). Die hier genannten Zahlenwerte sind aber nur ein Beispiel und bei jedem Programm anders! Übrigens ist der Tip der Redaktion offensichtlich heute nur noch bei einem IC-Fabrikat notwendig. Wie wir von den vielen Lesern gehört haben, die die Erweiterung nachgebaut haben, funktioniert sie in 99% aller Fälle wie im Schaltbild gezeichnet. Deshalb sollte zu diesem Kunstgriff nur in Ausnahmefällen gegriffen werden.

HARDCOPY FÜR SEIKOSHA GP 700A

Ich besitze einen Seikosha Printer GP 700A. Dieser ist mein Problem, ich besitze keine Hardcopy. Nun übersende ich Ihnen die Unterlagen für den GP 700A in der Hoffnung, daß Sie mir helfen können. Hoffentlich in Assembler, ich benutze den PIO-Kanal.
H.-J. Radiszewski,
Bochum

In letzter Zeit haben wir eine ganze Menge Anfragen nach Hardcopy-Routinen für neuere Seikosha Drucker erhalten. Wie aus den uns überlassenen Unterlagen hervorgeht, ist dabei gegenüber dem allgemeinen Epson-Standard die Wertigkeit der Nadeln genau vertauscht. Weiter wird die Anzahl der zu übermittelnden Grafikzeichen teilweise in dezimaler Form angegeben. Uns in der Redaktion liegt dazu leider keine Hardcopy-Routine vor, so daß wir leider nicht weiterhelfen können. Vielleicht aber kann einer der Leser helfen. Gerne würden wir eine derartige Routine auch veröffentlichen.

FALSCHE BUCHSTABEN AUF DEM DRUCKER

Ich habe ein kleines Problem mit meinem Star Drucker SG-10. Da dieser Epson-kompatibel ist, hoffe ich, daß Sie mir helfen können. Mit einem Programm sollen bei meinem Drucker drei Sonderzeichen definiert werden. Statt dessen schreibt der Drucker einige Zeichen auf das Papier, ab und zu macht er auch ein Formfeed oder ein Linefeed. Und zum Schluß sind meine neuen Zeichen immer noch nicht definiert. Was mache ich falsch?
Klaus-Jürgen Höltgen,
Essen

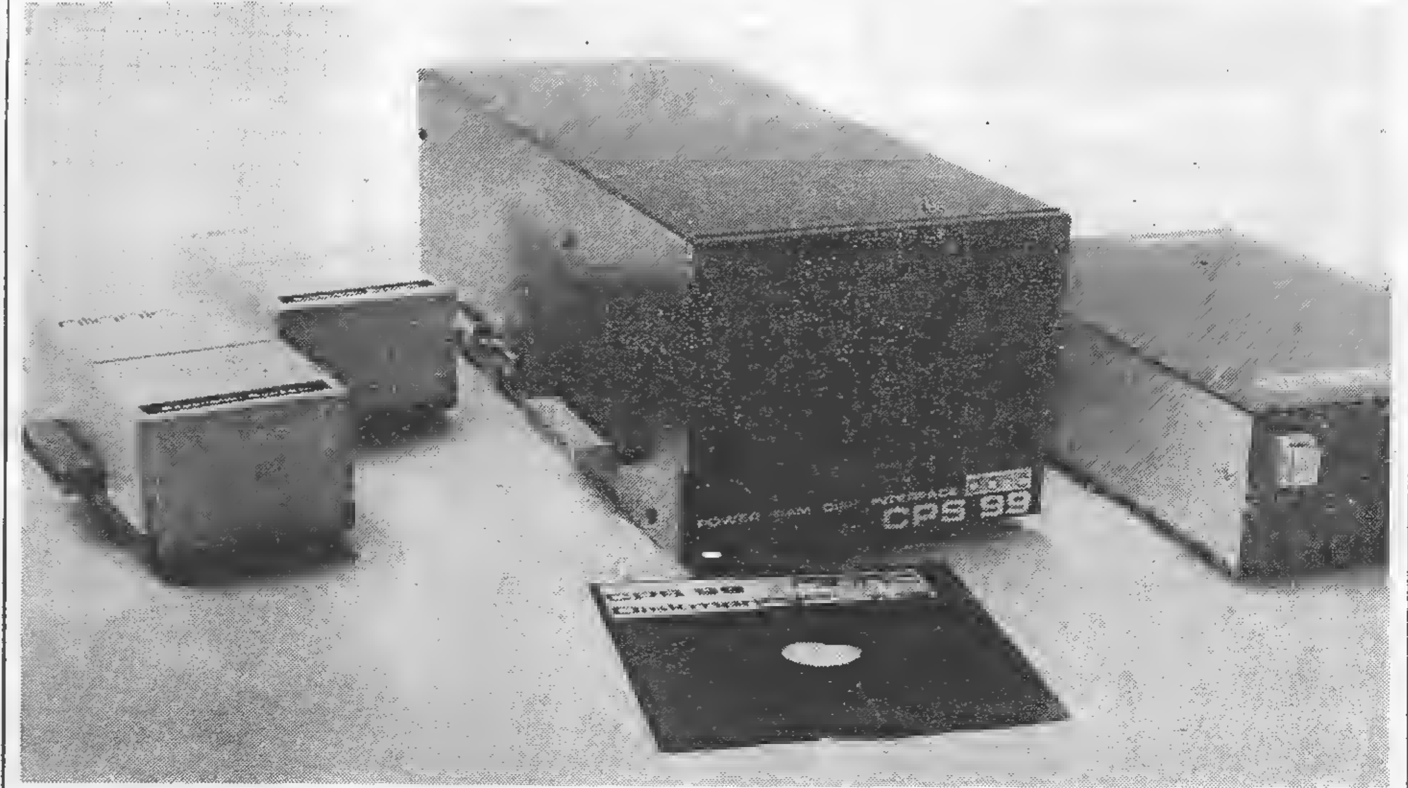
Die TI-Schnittstelle oder besser jede Drucker-Schnittstelle für den TI

99/4A schließt einen Datensatz mit einem Linefeed (Zeilenschub) und einem Carriage Return (Wagenrücklauf) ab. Kommen nun diese Zeichen zu einem Zeitpunkt, bei dem sie der Drucker nicht brauchen kann, kommt es zu Fehlfunktionen. Dies kann einmal bei bestimmten Steuerfunktionen, aber auch beim Grafikausdruck geschehen.
Hier gibt es grundsätzlich zwei Möglichkeiten: Einmal kann die Druckerdatei mit der Option eröffnet werden, daß diese Steuerzeichen unterdrückt werden.
Dies kann man durch Einfügen der Buchstaben ".CR" in den Dateinamen erreichen (Achtung, sollen beide Steuerzeichen unterdrückt werden, muß bei manchen Schnittstellen ".LF.CR" angegeben werden). Die andere Möglichkeit besteht darin, die Datensatzlänge möglichst groß zu wählen, z.B. VARIABLE 254. Passen dann alle Steuerzeichen nicht in eine PRINT-Zeile, so ist diese mit einem : abzuschließen. Dadurch wird der nächste PRINT-Befehl direkt an den vorherigen angeschlossen, ohne daß auf einen neuen Datensatz gegangen wird.

RATTERN DES DISKLAUFWERKES

Ich besitze einen TI 99/4A mit einem DS/DD Diskettenlaufwerk sowie seit drei Wochen das Editor/Assembler-Modul. Das Modul als solches arbeitet einwandfrei; jedoch zeigen Assembler und Editor einige "Mucken". Als ich ein selbstgeschriebenes Programm (bestehend aus 4 Teilen, Teil 1 mit 3 COPYs) assemblieren wollte, begann das Laufwerk nach ca. 20 Sekunden beängstigend zu rattern und das Licht des Disk-Controllers leuchtete ständig. Kurz darauf folgte die Fehlermeldung "DSR-ERROR-03C2 / I/O-ERROR CODE 0". Der fünfte Assemblierversuch gelang dann endlich. Als ich dann mit dem Editor Teil 1 des Quellcodeprogramms laden wollte, benötigte der Editor ca. 7 Minuten (Länge der Datei: 4 Blocks). Als ich mit dem Disk-Manager II die Diskette überprüfen wollte, erhielt ich die Fehlermeldung "SKETTE NICHT INITIALISIERT". Habe ich ein defektes Editor/Assembler-Modul Version 1.2, eine defekte Disk-Station (ich habe vor längerer Zeit das Türchen eine Zehntelsekunde zu früh geöffnet) oder eine defekte Diskette, obwohl auch bei anderen dieser Fehler auftritt?
Ralph Benzinger, Mannheim

Der hier geschilderte Fehler kann mehrere Ursachen haben. Einmal kann natürlich der Controller defekt sein, auch besteht die Möglichkeit, daß Laufwerk und/oder Disketten fehlerhaft sind. Das Rattern deutet darauf hin, daß der Controller bestimmte Sektoren auf der Diskette mehrfach suchen muß, bevor er diese einwandfrei lesen/schreiben kann. Um einen Sektor zu lesen, geht der Diskcontroller folgenden Weg: Anfahren der richtigen Spur mit dem Schreib-/Lesekopf. Wird



MEGA RAM NATÜRLICH VON ATRONIC

In der letzten Ausgabe hat leider der Druckfehlerteufel ganz besonders hart zugeschlagen: In der Überschrift zur Ankündigung des neuen Mega-Rams der Hamburger Firma Atronic wurde der Firmenname gleich mehrfach falsch geschrieben und so kann eine Verwechslung mit einer anderen Firma vorkommen.
Deshalb hier also noch einmal: Das Mega RAM ist eine Entwicklung der Fa. Atronic und wird auch von dieser vertrieben. Ebenfalls falsch war das dabei veröffentlichte Bild, es zeigte nicht das Mega-RAM von Atronic. Hier ein Foto.

nun innerhalb von 15 Umdrehungen der Diskette der gewünschte Sektor nicht gefunden (normalerweise ist das beim ersten Mal der Fall) wird der Kopf auf Spur 0 zurückgefahren und die gewünschte Spur neu angefahren (das ist das Rattern) und wieder 5 Umdrehungen lang versucht, den Sektor zu lesen. Der letzte Vorgang wiederholt sich 7mal, bevor ein ERROR ausgegeben wird. Lesefehler treten natürlich bei defekten Laufwerken auf (eine Überprüfung kann jede gute Computerwerkstatt mit Laufwerksmeßplatz vornehmen). Eventuell können auch schlechte Kabel-/Steckverbindungen der Grund sein. Dabei muß natürlich vorausgesetzt werden, daß das Laufwerk regelmäßig mit einer Reinigungsdiskette gereinigt wird. Etwas merkwürdig ist allerdings der Error-Code und das offensichtliche Überschreiben des Sektors 0 (Diskette nicht initialisiert). Hier könnte einmal der Kopf des Laufwerkes hängen, es ist aber auch eine defekte Speichererweiterung denkbar.

BASIC-PROGRAMME ZERSTÖRT

Das notwendige Teilen eines zu lang gewordenen Dateiprogramms habe ich durch Umkopieren, erhebliches Tilgen von Programmzeilen mit "ERASE" und entsprechenden Änderungen bewerkstelligt. Leider quittierte dies mein TI damit, daß er Fehler bei der Programmaufzeichnung auf Diskette vornahm. Fehler, die sich wie eine Krebswucherung ausbreiteten und das Programm selbst von innen heraus zerstörten. Bei den ersten Programmausdrukken war dies kaum feststellbar, beim Programmablauf dehnten sich die Programmzeilen plötzlich mit einem sinnlosen zertrümmerten Mosaik gummiartig über zwei Bildschirmlängen, dann stieg die Konsole aus, füllte den Bildschirm mit einem blödsinnigen Muster und war nur mehr durch Ausschalten zu bändigen. Ich habe daraufhin den Computer überprüfen lassen und startete wieder hoffnungsvoll: Die gleiche Erscheinung, zumindest mit den beschädigten Programmen war nichts mehr anzufangen. alle Versuche einer Rekonstruktion scheiterten. Ein TI-Fan sagte mir nun, ihm wäre das auch schon zweimal passiert. Hier läge wohl ein Fehler im Betriebssystem vor, der bei bestimmten, selten auftretenden Kombinationen dieses Programm-Verwirrspiel verursacht, das selbst die einzelnen Statements in Trümmer zerlegt.
Der Ansatz für diese Erscheinung liegt nach meiner Beobachtung besonders in den CALL Unterprogrammen sowie in den REM-Zeilen. TI weiß davon aber nichts. Ist Ihnen oder Ihren Spezialisten etwas davon bekannt? Und wie kann man solche Programm-Selbstzerstörungen vermeiden?

Peter Grimm, Starnberg

Interessanterweise bekommen wir solche Meldungen immer gerade nach gewitterreichen Wochen. Von einem Fehler im Betriebssystem ist uns diesbezüglich nichts bekannt, vielmehr ließ sich bei unserer Arbeit der Fehler immer relativ genau lokalisieren: Bedingt durch den besonderen Aufbau des Basic-Programms im Speicher des TI 99/4A kann die Änderung einzelner Bytes im Speicher gravierende Fehler hervorrufen. Diese unerwünschten Änderungen können z.B. durch sehr kurze Stromausfälle hervorgerufen werden, oder auch durch ein versehentliches Herausnehmen der Diskette während des Abspeicherns. Je nachdem, wieviel bzw. was in dem Programm schon zerstört ist, kann man versuchen, das Programm über MERGE zu retten. Es bietet sich auch ein LIST auf Diskette (wenn sich der Computer dabei nicht aufhängt) an, mit der anschließenden Rückwandlung über ein CONVERT-Programm (die falschen Zeilen können vorher z.B. mit dem TI-Writer editiert werden.

UNSCHÖNE BILDSCHIRMAUSGABEN BEI GPLLNK

Ich habe eine Frage, bei der ich einfach nicht weiterkomme: Es handelt sich dabei um GPL-Routinen, z.B. der Berechnung eines einfachen Sinus oder Cosinus. Bei der Berechnung habe ich einen geeigneten Wert in den FAC gelegt, das GPL-Statusbyte gelöscht und die Routine mit
BLWP §GPLLNK
DATA >002E
aufgerufen. Ergebnis: Es steht zwar der richtige Wert in FAC, jedoch zeigt der Bildschirm in der Mitte der oberen Zeile unsinnige Zeichen an. Natürlich habe ich weiter probiert und in der mir zur Verfügung stehenden Literatur gesucht. In einer amerikanischen Veröffentlichung findet sich nun ein schwacher Hinweis, daß man den Wert nicht nur auf FAC, sondern auch in das VDP-RAM und die Anfangsadresse dieses Fließkommawertes im VDP-RAM in die Adresse >836E. Angestachelt durch diesen vagen Hinweis gelang es mir, das Anzeigen der sinnlosen Zeichen am Bildschirmkopf zu vermeiden. Meine Frage ist natürlich, warum das so funktioniert, oder besser, wie man es richtig macht? Noch ein Problem ergibt sich im Bitmap Mode. Da werden durch das Aufrufen der GPL-Routine irgendwelche Bytes in der Color-Table belegt.

Robert Hundt, Ottobrunn

Die mathematischen GPL-Routinen benutzen den Basic-Stack. Der Zeiger auf das Ende von diesem Stack befindet sich auf >836E. Wenn man also nicht vom Basic (oder Extended Basic) heraus das Assemblerprogramm aufruft, welches auf die GPL-Routinen zugreift, muß man diese Speicherstelle mit einem Zeiger belegen, der auf irgendeinen freien Bereich im VDP-RAM zeigt. Den Wert der Zahl selber aber braucht man nicht in das VDP-RAM zu schreiben. Weiter sichern diese GPL-Routinen einen Teil des Scratch-Pad RAMs in den VDP ab Adresse >03C0. Dagegen läßt sich nichts machen, denn dieser Wert ist im GROM festgelegt. Im Bit-Map Mode kann man nun diesen Teil des VDP-RAMs vor Aufrufen der GPL-Routine sichern und danach wieder hineinschreiben, ein kleines Flackern aber bleibt.

MEHR NACHKOMMASTELLEN MIT 'PRINT USING'

Wie wohl bekannt, rechnet der TI mehr Nachkommastellen aus, als er mit PRINT anzeigt. Es ist jedoch möglich, mit PRINT USING "##.######
######":variable zwölf Nachkommastellen anzuzeigen.

Eckstein

Die nächste TI-REVUE am 27. Juni

DISK-INIT & DISK-EDIT SOFTWARE FÜR DISKETTENLAUFWERKE MADE EIGENBAU

Nachdem Texas Instruments mit der Produktion des TI 99/4A aufhörte, standen einige Leute vor dem Problem der Erweiterung des Computers. Manche warteten die Zeit ab, bis andere Firmen gewisse Teile nachbauten, und andere versuchten sich im Selbstbau dieses Zubehörs. Es ist auch vielen gelungen, eine komplette Peripherie-Box nachzubauen, aber sie hatten dann noch nicht die Software zum Initialisieren von Disketten und gewisse andere Routinen. So wurden von der Michael Franz Elektronik Ende 1984, Anfang 1985 mehrere Programme dafür vorgestellt und vertrieben. Diese Programme heißen DISKINIT und DISKEDIT und sind beide in Assembler geschrieben. Für diese Routinen sind eine Speichererweiterung und mindestens ein Laufwerk notwendig.

DISKINIT

Wird die Diskette mit der DISKINIT-Routine in das Laufwerk 1 gegeben und das EX-Basic angewählt, so wird automatisch die LOAD-Routine eingelesen. Mit Hilfe dieser Routine werden ein neuer Zeichensatz und ein weiteres Hilfsprogramm zum Einlesen der Utilities geladen. Dieses Hilfsprogramm gestattet es dann, die einzelnen Unterroutinen BACKUP und CATALOG, DELETE und FORMAT, RENAME und PROTECT einzuladen und sie zu starten.
Als erste Routine wäre da der Programmteil FORMAT zu nennen. Er ermöglicht es, eine neue Diskette zu formatieren. Bei diesem Programmteil kann man es sich aussuchen, mit wieviel Tracks bzw. Spuren man eine Diskette ausstatten will. Diese Anzahl kann zwischen 35 und 80 liegen. (Je nach Laufwerk und Controller.) Zudem kann man einseitige oder zweiseitige Disketten nutzen, wobei es wieder auf die Art der Laufwerke ankommt.

SPEECH-SYNTHESIZER (SS)

Will man in einem Programm den SS benützen und das Programm nicht ändern, ist dieser Tip genau das richtige. Hat man keinen SS angeschlossen, entstehen lange Wartezeiten, die vermieden werden können. Am Anfang des Programmes CALL PEEK (28672, SYS). Ist der Wert von SYS=96, dann ist der SS angeschlossen und es kann gesprochen werden. Bei Sprachausgabe im Programm ist nur noch zu schreiben: IF SYS=96 THEN CALL SAY ("HELLO").

Für eine einseitige Diskette mit 40 Spuren benötigt der Computer ca. 53 Sekunden. Danach ist die Diskette auf den Namen BLANK 'getauft', den man aber jederzeit mit der Unterroutine RENAME wieder umändern kann. Diese Unterroutine ermöglicht es dann, sowohl Disketten- als auch Dateinamen umzuändern.
Wählt man die Unterroutine BACKUP an, kann man auch kopiergeschützte Disketten kopieren, wenn man Besitzer zweier Laufwerke ist. Dieses Programm geht Sektor für Sektor vor, und so kann man ohne direktes Einlesen eines Programmes dieses vervielfältigen. Bei einer Single Sided/Single Density-Diskette benötigt ein Kopiervorgang 162 Sekunden, wobei noch 14 Sekunden für das Initialisieren der Diskette eingerechnet sind. Allerdings ist im Programm ein kleiner Fehler, der das Kopieren nur dann erlaubt, wenn man gleichzeitig die Diskette formatieren läßt und nicht wie abgefragt ohne Initialisieren. Dieser Fehler hat keine Auswirkung auf die Kopie, aber man muß jedesmal 14 Sekunden länger warten.
Bei dem Programmablauf des Kopierens werden immer 4 Tracks (= 36 Sektoren) eingelesen und dann auf der anderen Diskette abgespeichert.
Will man wissen, welche Programme auf einer Diskette sind, kann man dies über die Rounte CATALOG erfahren.
Mit Hilfe dieses Programms kann man eine Diskette katalogisieren, wobei es aber an einer Druckerausgabe fehlt.
Zudem sind in dem DISK-UTILITY-Paket noch zwei Programme enthalten die es erlauben, Files zu löschen und Files gegen unbeabsichtigtes Löschen zu sichern.
Nun zu dem anderen Programmpaket mit dem Namen DISKEDIT.
Mit diesem Programm kann man die Sektoren direkt anschauen und auch manipulieren.
Der Start des Programms ist identisch mit dem Start der vorher genannten Routinen.

DISKEDIT

Wenn das Programm gestartet ist, steht dem User ein Interpreter zur Verfügung, mit dem man ein Laufwerk von dreien einschalten, Spuren der Diskette lesen, verändern und auch wieder abspeichern kann.
Man soll auch mit diesem Programm die Sektoren einer Diskette auf dem Drucker auflisten können, aber dies konnte ich nicht mit meinem Drukker und der TI-PIO nachvollziehen.
Demgegenüber steht der gute Interpreter, der es auch ermöglicht, Befehlsfolgen immer wieder zu durchlaufen.
Beispiel:

L2;B0	Schalte Laufwerk 1 an und gehe zu Sektor 1
D;+;	Drucke den ersten Sektor auf dem Bildschirm aus, erhöhe den Zähler und gehe wieder zum Anfang, sprich zeige den nächsten Sektor an.

Dieses kurze Beispiel zeigt schon die Nützlichkeit dieses Interpreters.
Abschließend kam ich zu dem Urteil, daßdiese Programme sehr nützlich, einfach zu handhaben und die Anleitungen ausführlich und gut verständlich geschrieben sind.

HL

DISKETTEN UNTER'S HEMD GESCHAUT

Da die meisten User ihren TI 99 ausbauen, anstatt zu verkaufen und sich eine RAM-Erweiterung und ein Diskettenlaufwerk zulegen, werden diese sich notgezwungen mit der Diskette etwas näher beschäftigen müssen.
Gehen wir zunächst in groben Zügen über die Diskette. Als erstes finden wir auf Sektor 0 die allgemeine Disketteninformation.
Auf Sektor 1 befindet sich eine sogenannte LINK MAP (auf Deutsch: Verzweigungstabelle). Diese dient zum Herstellen des Kataloges und zum Laden der Programme, doch dazu später!
Von Sektor 2 bis (normalerweise) 21 ist Raum für die einzelnen Programmnamen und deren Informationen.
Ab Sektor 22 stehen nun also die eigentlichen Programme; diese können maximal, auf einem Laufwerk mit einfacher Dichte, einseitig beschrieben, bis Sektor 167 stehen.
Als kleine Grafik können wir uns dies folgendermaßen vorstellen:

Sektor 0 : Disk-Infos
Sektor 1 : Link Map
Sektor 2 bis 21: Raum für Programminfos (was, wieviel, wo)
ab Sektor 22 bis 167: Programme, Daten ...

Hinweis:
Alle Sektorangaben in Hexadezimal!

1) SEKTOR 0:

Wenn man eine Diskette mit dem Manager Modul oder einem anderen Programm neu initialisiert, erhält diese einen Namen, der in den Bytes 0 – 9 von Sektor 0 steht.
Danach folgt in den Bytes A und B (Hexadezimal!!!) die Anzahl der gesamten Sektoren (auf Laufwerken mit einfacher Dichte/einseitig beschrieben: 168).
Danach folgt als Byte C die Anzahl der Sektoren pro Spur.
Die Diskette hat 40 Spuren mit je 9 Sektoren (Probe: 40 x 9 = 360).
Nach dem Wert folgen drei ASCII-Werte, die zusammengesetzt "DSK" ergeben.

20 = UNGESCHÜTZT

Für was dies gut sein soll, wissen nur die Götter bzw. die von TI!
... Dann folgt eine Bytefolge, die dem Computer angibt, ob die Diskette geschützt ist, auf wieviel Spuren die Diskette initialisiert wurde, ob die Diskette einseitig oder zweiseitig ist und ob die Dichte einfach oder doppelt ist. Also, hat man eine Diskette, bei der folgendes auf diesen Bytes steht, ist diese ...
20=ungeschützt (kann mit Diskmanager kopiert werden)
28=hat 40 Spuren (=28 hex!)
01=einseitig bespielbar
01=und hat einfache Dichte!
Die Diskette ist geschützt, wenn anstatt dem Byte 20 ein Wert von 50 (ASCII= "P") dort steht!
Ab hier folgt bis Byte 38 auf diesem Sektor nichts mehr Sinnvolles, wird aber von da ab um so schwieriger ...
Diese Bits sind von links nach rechts von 0 – 7 durchnumeriert. Jedes Bit besitzt auch einen entsprechenden Dezimalwert und auf der Diskette einen Sektor, den das Bit überwacht.

DER BITMAP:

Nein, das ist kein VDP-Modus für die Diskette, sondern diese Bytes legen fest, welche Sektoren auf einer Diskette belegt und welche unbelegt sind.
Jetzt werdet Ihr vielleicht denken, daß der Computer 360 Bytes (oder ebenso viel Bytes, wie die Disk Sektoren hat) daran verschwendet.
Da der TI aber wirklich nicht so verschwenderisch umgeht, kontrolliert jedes Byte maximal 8 Sektoren gleichzeitig!
Wie das geht?
Byte 38 kontrolliert die Sektoren von 0–7.
Byte 39 kontrolliert die Sektoren von 8–F.
Und so weiter ...
0–7, das kommt doch irgendwie bekannt vor? Genau, das hat etwas mit den Bits zu tun. 8 Stück an der Zahl, die jedes Byte besitzt!

Also: Byte 4C kontrolliert die Sektoren von A0 bis einschließlich A7.
Da aber die Bitmap Adresse 4C auch noch einen Wert enthält, sagen wir mal 7B, können wir genau feststellen, welche Sektoren in dem Block von Sektor A 0 bis A7 benutzt werden.
7B ergibt dezimal 123. Diesen Wert aufgeschlüsselt in unsere Bit Tabelle, erhält man dann binär:

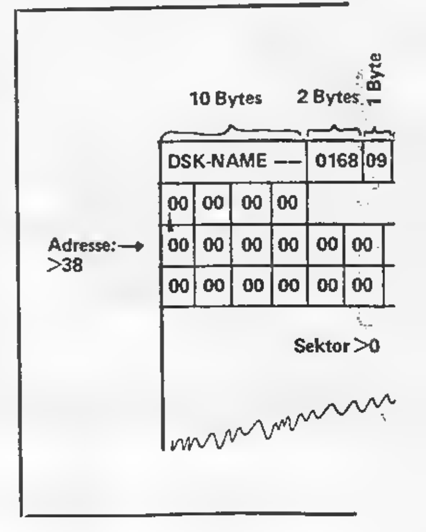
BIT	0	1	2
DEZ:	128	064	032
ges.:	0	1	1
Sektor:	0A7	0A6	0A5

Jedes gesetzte Bit des Wertes aus der Adresse 4C bedeutet, daß der jeweilige Sektor belegt ist.
In unserem Beispiel:
Sektoren: A0, A1, A3, A4, A5, A6 sind belegt
Sektoren: A2 und A7 hingegen nicht!
Soweit über Sektor 0, und

BIT	0	1	2	3	4	5	6	7
Dezimalwert:	128	064	032	016	008	004	002	001
Sektor:	x+7	x+6	x+5	x+4	x+3	x+2	x+1	x

Das x ist der Sektorblock, der von dem Byte überwacht wird. Es errechnet sich aus der Adresse des Bytes minus der Adresse des BITMAP Startes (also 38). Multipliziert man das ganze mit 8, erhält man den Block.
Nicht verstanden? – Hier ein Beispiel:
Byte 4C von Sektor 0 kontrolliert welchen Sektorblock?
Man rechnet am besten die Werte ins Dezimalsystem und erhält für 4C = 76 und für 38 = 56.
Nun werden die gefundenen Dezimalwerte subtrahiert: 76 – 56 = 20.
Dieser Wert wird nun mit 8 multipliziert und wir erhalten 160 (dez); dies entspricht A0 (hex).

nun geht es weiter mit der LINK Map auf Sektor 1.
Zuvor aber noch eine kleine Grafik über den Aufbau von Sektor 0 ...



2) SEKTOR 1:

In Sektor 1 steht die LINK MAP einer Diskette. Diese Tabelle regelt den Katalog und die Routinen zum Laden. Diese Link Map baut sich folgendermaßen auf: Nehmen wir an, es wäre ein Programm mit dem Namen "PGM1" auf der Diskette gespeichert, so würde in dieser Tabelle 0002 stehen, da die Programminformationen in Sektor 2 enthalten sind. Würde man nun ein anderes Programm mit dem Namen "OS-SOFT" auf die Disk saven, so würde sich die LINK MAP so verändern:

0003 0002 0000 . . .

3	4	5	6	7
016	008	004	002	001
1	1	0	1	1
0A4	0A3	0A2	0A1	0A0

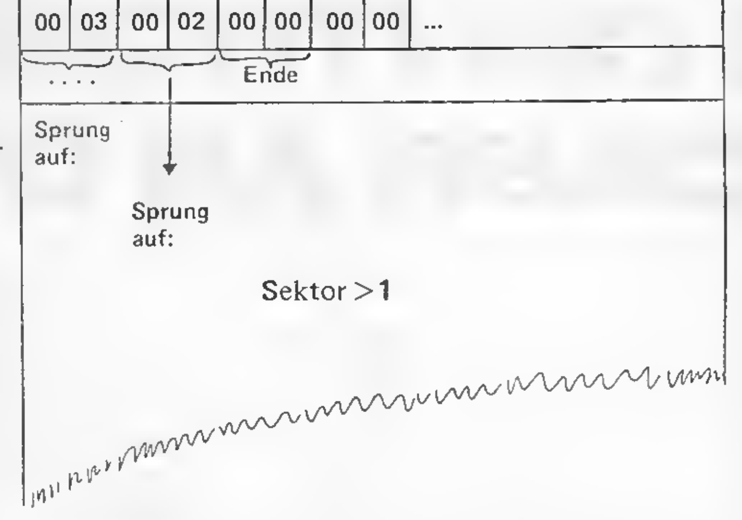
Wieso das? Da der Controller die Programme für den Katalog alphabetisch ordnet, kommt nun das Programm mit dem Namen "OS-SOFT" vor "PGM1"!

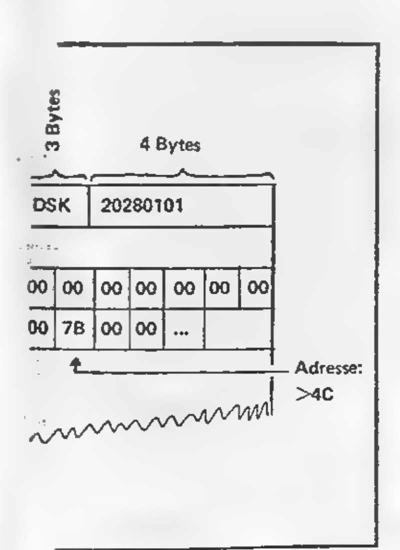
Die 0003 zeigt an, daß die Informationen über das Programm ("OS-SOFT") in Sektor 3 sind.

Beim Katalogisieren würde dann der Computer so vorgehen:
- trifft auf Sektor 0 – liest Disk Namen/freie Sektoren
- trifft auf Sektor 1 – verzweigt zu Sektor 3
- liest in Sektor 3 den Programmnamen und andere Daten
- springt zurück auf Sektor 1 – verzweigt zu Sektor 2
- liest in Sektor 2 den Programmnamen . . .

Dies führt der Computer so lange aus, bis er auf ein 0000 in Sektor 1 trifft!

Auch dazu eine kleine Grafik:





3) SEKTOREN 2 bis 21:

Auf diesen Sektoren befinden sich die Programminformationen für den Computer bzw. für den User.

Pro Programm bzw. Datei wird ein Sektor benötigt, der wie folgt aufgebaut ist:
- Filename
- Fileart (Programm oder Datei)
- Anzahl der Sektoren, die das File benutzt
- Anzahl der Bytes, die im letzten Sektor vom Programm sind.
- Datensatzlänge
- Anzahl der Datensätze
- Wo das File auf der Diskette steht (Startsektor)

Beginnen wir mit dem Programmnamen:

Dieser darf maximal 10 Bytes lang sein und steht auf den Bytes 0 – 9!

Auf den Bytes A bis F stehen die Spezifikationen über die Art der Files.

Die Bytes A und B haben meiner Meinung nach keine Bedeutung . . .

Byte C hingegen wird wieder in Bits unterteilt:

BIT:	VERWENDUNG:
0	?
1	1:Internal oder . 0:Display Format
2	?
3	1:Write Protected 0:nicht Write Protected
4	?
5	?
6	?
7	1:Variable Record's 0:Fixed Record's

Wer mehr wissen möchte, schaut im E/A Handbuch unter File Management nach.

Byte D gibt die Anzahl der Datensätze pro Sektor an.

Falls es ein Programmfile ist, steht hier 00!

In den Bytes E und F sind die Anzahl der Sektoren, die das File benötigt, festgehalten!

Nun folgt in Byte 10 die Anzahl der Bytes, die vom letzten Sektor eingelesen werden.

Das Byte 11 gibt an, wieviel Bytes pro Datensatz eingelesen werden. (Bei z.B. OPEN Statement als Fixed xx definiert.)

Byte 12 gibt die maximale Anzahl der Datensätze im File an.

. . . Von Byte 13 bis 1B stehen nur 00's.

Danach folgt der sogenannte BLOCK LINK, der den Startsektor des Programmes angibt und dorthin verzweigt!

Dieser Block Link ist wie folgt aufgebaut:

Byte 1C bis 1F:

Diese Bytes geben den ersten Sektor an, von wo ab das Programm auf der Diskette steht. Es kann vorkommen, daß ein Programm in mehrere Teile geteilt wird und mehrere ENTRYs besitzt.

Byte 1C:

enthält das niederwertige Byte vom Sektor.

Byte 1D:

Bits 0 bis 3: enthält das höherwertige Byte vom Sektor (für Sektoren über FF!).

Bits 4 bis 7: enthalten das niederwertige Nibble von der Anzahl der Sektoren minus 1!

Byte 1E:

Bits 0 bis 3: enthält das höherwertige Nibble vom niederwertigen Byte der Sektorenanzahl.

Bits 4 bis 7: höherwertiges Nibble der Sektorenanzahl (–1 nicht vergessen!)

Dieses Nibble wird natürlich nur benutzt, wenn die Sektorenzahl größer FF ist!

Achtung:

BITFOLGE –

7	6	5	4	3	2	1	0BIT
128	64	32	16	8	4	2	1DEZ

Diese Liste steht normalerweise von Sektor 2 bis 21 (hex). Es kann aber auch vorkommen, wenn zuviele Programme auf der Diskette sind, daß ein solcher Sektor mitten zwischen den Files steht.

Dies kann man schon im Sektor 1 erkennen, wohin die Link Map zeigt!

4) SEKTOREN AB 22:

Ab Sektor 22 stehen die Files (normalerweise!). Gehen wir auf den Aufbau eines Basic Programms auf der Diskette näher ein.

Diese sind auf der Diskette genauso gespeichert, wie sie auch im Speicher stehen!

Das heißt, sie sind in 2 Teile unterteilt:

Der erste Teil ist der LINE NUMBER TABLE, der zweite Teil sind die Programmzeilen.

Vor dem Line Number Table (LNT) stehen noch einige Informationen für den Computer, die wissenswert sind:

Bytes 0 und 1:
sind für den Basic Loader gedacht, damit dieser weiß, daß nun korrekte Daten kommen.
Bytes 2 und 3:
geben die Adresse vom Ende des LNT an. Folglich geben die
Bytes 4 und 5:
Anfangsadresse des LNT an . . .
Bytes 6 und 7:
gibt die höchste Adresse an, an der das Basic-Programm auftaucht!

Dies sind alles Speicher-Zeiger und dienen dazu, daß das Programm richtig wieder eingelesen wird. Danach folgt der LNT, der jeweils mit der Zeilennummer und deren RAM Adresse versehen ist.

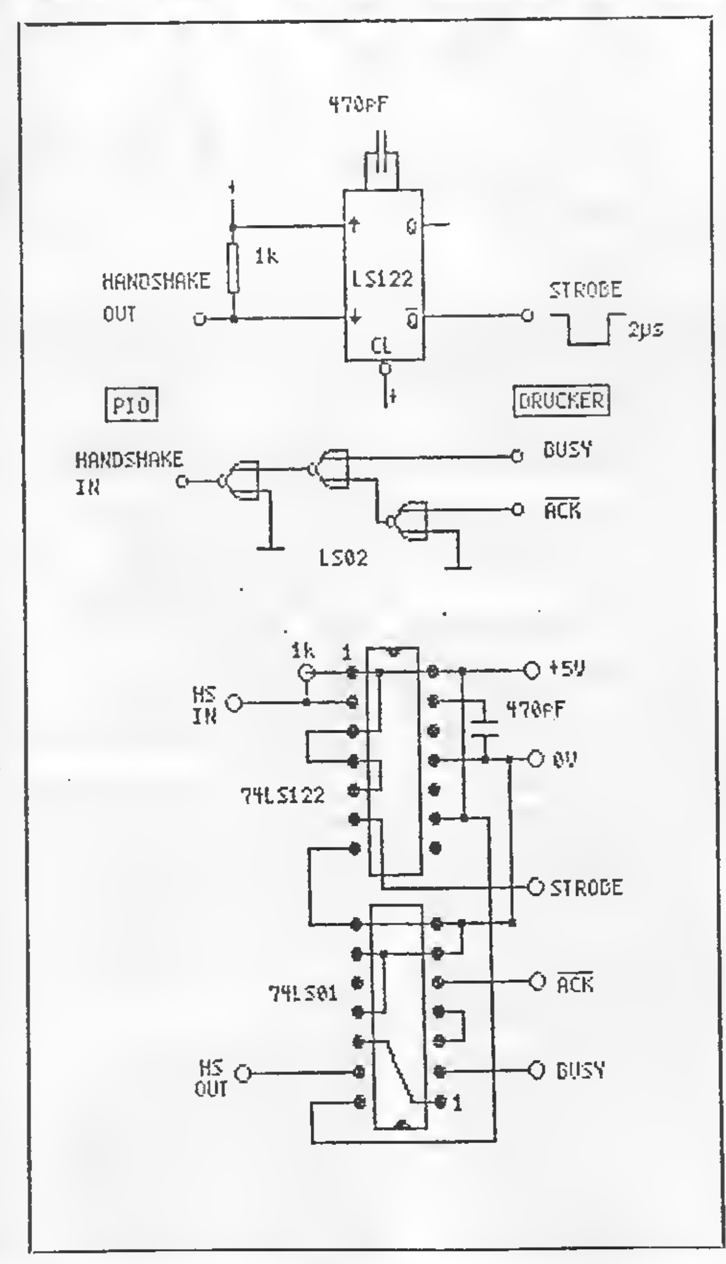
Ist der LNT vorüber, folgt das eigentliche Programm, welches aus den einzelnen Zeilen zusammengesetzt wird. Jede Zeile ist nach diesem Schema aufgebaut:
– die Länge der Zeile
– das Token
– die Daten für das Token
– am Ende der Zeile 00.

Dies soll als kleine Einführung dienen. Ich hoffe, mit diesem Bericht etwas Licht ins Dunkel der Diskette gebracht zu haben.

Oliver Siffrin

ALTE CENTRONICS-DRUCKER AN DIE TI-PIO

Wie auch Herr Leititis in der Märzausgabe Ihrer Zeitschrift bemerkte, hatte ich die selben Probleme beim Anschluß eines alten Centronics-Drucker (Modell 779) an meine TI-PIO.
Bei einem Epson-Drukker, den ich vorher benutzte, hatte ich diese Probleme nicht, wie sie nun auftraten. Daraufhin untersuchte ich die PIO-Schnittstelle mit einem Oszillographen und besorgte mir zusätzlich einen Schaltplan meines Drukkers. Nach dieser genaueren Untersuchung wurde mir nun klar, warum die Kommunikation nicht funktionierte: Der TI erwartet eine Antwort vom Drucker auf die abfallende Flanke des STROBE-Signals, während der Drucker erst auf die absteigende Flanke dieses Signals reagiert und das BUSY-Signal setzt. Bei Epson-Druckern ist das anders, diese antworten bereits nach Empfang der abfallenden STROBE-Flanke mit dem BUSY-Signal.

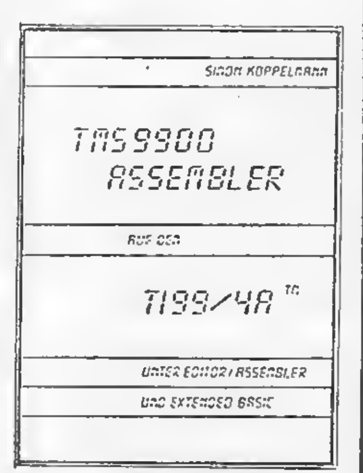


Nach diesem Ergebnis der Analyse baute ich eine kleine Zusatzschaltung auf einem Lochrasterstreifen in den Drucker ein, die dafür sorgt, daß der Drucker nach ca. 2 Mikrosekunden eine ansteigende Flanke des STROBE-Signals erhält. Die Verzögerung ist erforderlich, einfaches Invertieren des STROBE ist nicht ausreichend.
Zusätzlich realisierte ich noch eine ODER-Verknüpfung zwischen den Antwortsignalen BUSY und ACKNOWLEDGE des Druckers, da die TI-Schnittstelle nur einen Handshake-Eingang besitzt (die Bedeutung der Spare-Bits ist leider nicht dokumentiert).
Bei den Schaltskizzen und dem Aufbauplan meiner Schaltung muß ich noch darauf hinweisen, daß es unter Umständen Probleme bei der Terminierung der Eingangssignale geben kann, welche zu Funktionsstörungen führen können. Bei einem alten Drucker ist der STROBE-Eingang statt mit 1 kOhm nur mit 470 Ohm termiert. Dadurch wird die zulässige Ausgangslast der 74LS-Bausteine in der PIO überschritten!

F. Behnisch

TMS 9900 ASSEMBLER AUF DEM TI 99/4A

Das ca. 270 Seiten starke Buch von Simon Koppelmann beschäftigt sich mit dem 16-Bit Prozessor TMS 9900, mit der Programmierung und dem Einsatz dieses Prozessors im TI 99/4A.
Es handelt sich dabei mehr um ein Handbuch zu diesem Prozessor als ein Lehrbuch und fordert somit schon eine gewisse Basis an Wissen über die Programmierung des TMS 9900. In diesem Buch werden die grundsätzlichen Konzeptionen von CRU (Communication Register Unit) und DSR (Device Service Routine) klar dargelegt und erklärt, welche sogar umfangreicher sind als die Erklärungen, die dazu im TI-Assembler-Handbuch zu finden sind.
Auf den ersten 15 Seiten findet man eine Erklärung zur Benutzung des EDITOR-ASSEMBLERs im EDIT-Modus. Hier wird das Eingabeformat genau erklärt, welches dann auch für die Befehlsliste nützlich ist.
Vor der eben schon genannten Befehlsliste findet man noch die Aufgliederung der Befehle in die einzelnen Formate.
In der alphabetisch geordneten Befehlsliste findet man jeden Befehl ausführlich erklärt. Jeweils mindestens ein Beispiel dazu wird genannt. Diese Befehlsliste umfaßt nicht nur die Befehle des TMS 9900, sondern auch die Befehle des TI-Assemblers.
Die nun angesprochenen Teile des Buches umfassen die ersten 90 Seiten, wobei man anmerken kann, daß die fehlenden Zeilenangaben in der Inhaltsangabe manchmal die Suche eines bestimmten Befehls schwieriger gestalteten.
An diesen ersten einführenden Teil schließt sich nun der Teil an, in dem die CRU- und DSR-Handling genau beschrieben und erklärt wird.
Diese Beschreibung und Erklärung erfolgt am Aufbau dieser Teile und an Beispielen dazu. Zusätzlich sind auch umfangreiche Erläuterungen zum VDP und GROM in diesem Buch enthalten.



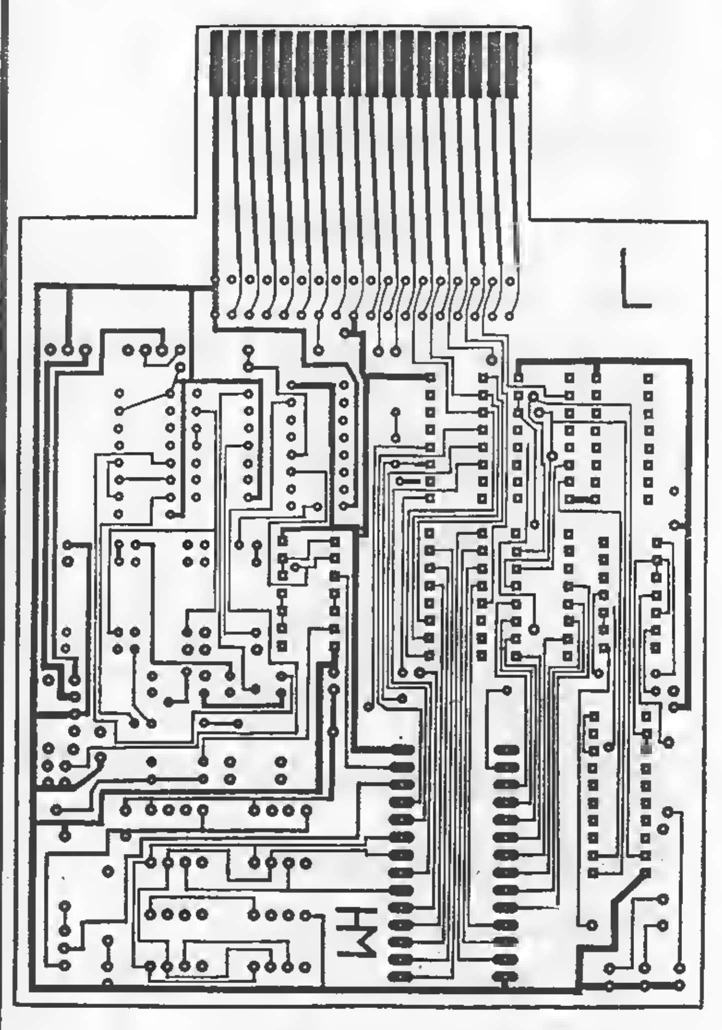
All diese, in dem zweiten Teil zu findenden, Routinen sind ausführlicher besprochen als zum Beispiel im ASSEMBLER-Handbuch oder in dem 16 Bit Mikroprozessor-Kursbuch.
Der dritte Teil besteht hauptsächlich aus mehreren Anhängen zur Programmiertechnik, zur Fehlerfimdung und aus 8 Beispielen der Programmierung des TMS 9900.
Abschließend kann man sagen, daß dieses Buch eine gute Hilfe zur Programmierung ist, da man die wichtigsten Dinge des TI 99/4A in einem handlichen Buch zusammengefaßt und erweitert hat.

Heiko Liebald

Bezugsquelle:
Simon Koppelmann
Judenpfad 7
5000 Köln 50

ERWEITERUNGS-PROGRAMM ZUR EPROMMER-PLATINE

Das in Ex/Basic geschriebene Programm beinhaltet einige Erweiterungen zum Assembler-Programm aus TI-REVUE 7/85.
Das Assembler-Programm wird geladen und ist Bestandteil des Programms. Der vom Eprom in einen anderen Speicherbereich (z.B. D000-D400 > 2K >2716) abgelegte Inhalt kann auf dem Bildschirm sichtbar oder auf dem Drucker ausgedruckt werden. Die Ausgabe erfolgt in Hex-Zahlen.
Ferner kann der Speicherbereich auf Disk abgespeichert und wieder eingelesen werden. Beim Einlesen wird die Datei sofort wieder in den vorher festgelegten Speicherbereich geladen.



Ferner können Eprom-Inhalte geändert und neu geschrieben werden.
Es können somit irgendwelche Eproms geschossen werden, sofern man die Hexdump-Listings besitzt. Oder man kann jeden Eprom-Inhalt aus anderen Geräten ausdrucken, ändern oder als Master-Eprom auf Diskette abspeichern.

Bernd Eder

LISTINGS

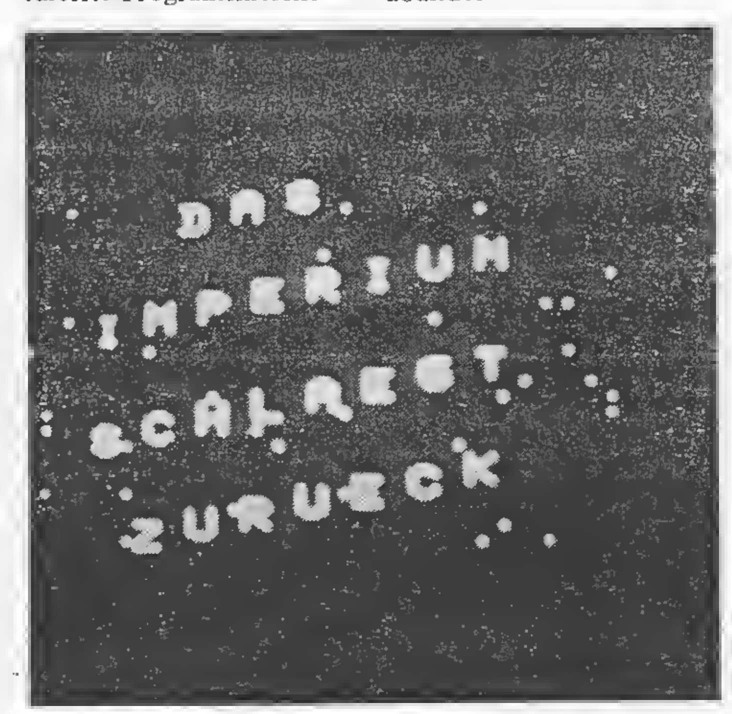
```
10 !************************
11 !*   ERWEITERUNG ZUM     *
12 !*    EPROMMER AUS       *
13 !*   TI/REVUE 7/85       *
14 !*                       *
15 !*    Copyright  by      *
16 !*     Bernd Eder        *
17 !*                       *
18 !*  Benoetigte Geraete   *
19 !*   TI99/4A  Konsole    *
20 !*    Extended Basic     *
21 !*   Eprommer Platine    *
22 !*  Speichererweiterung  *
23 !*     Disk Laufwerk     *
24 !*     even. Drucker     *
25 !*                       *
26 !*   Speicherbelegung    *
27 !*     4082 Bytes        *
28 !*                       *
29 !*******************be**
100 CALL CLEAR :: CALL INIT
:: ZA=Ø
110 CALL SCREEN(13):: FOR Z=
1 TO 12 :: CALL COLOR(Z,16,1
3):: NEXT Z :: DISPLAY AT(4,
5)ERASE ALL BEEP:"*** E P R
O M mer ***" :: CALL COLOR(Ø
,16,13)
120 DISPLAY AT(9,2):"1 >> RA
M-INHALT AUSDRUCKEN": :" 2 >
> RAM-INHALT AENDERN": :" 3
>> EPROMMER LADEN": :" 4 >>
DATEI LADEN"
130 DISPLAY AT(22,5):"BITTE
ZIFFER DRUECKEN" :: CALL KEY
(Ø,K,S):: DISPLAY AT(22,11)S
IZE(6):"      " :: IF S=Ø TH
EN 130 ELSE IF K<49 OR K>52
THEN 130
140 ON K-48 GOTO 170,320,150
,440
150 DISPLAY AT(12,5)ERASE AL
L BEEP:"EINEN MOMENT BITTE"
:: IF ZA=Ø THEN CALL LOAD("D
SK1.EPROM_O"):: ZA=1
160 CALL CLEAR :: CALL SCREE
N(5):: FOR Z=1 TO 12 :: CALL
 COLOR(Z,16,5):: NEXT Z :: C
ALL COLOR(Ø,16,5):: CALL LIN
K("EPROM"):: CALL CLEAR :: G
OTO 110
170 CALL ADD(ADA,ADE):: ANF=
ADA :: DISPLAY AT(14,3):"DRU
CKERAUSGABE >J/N< N"
180 ACCEPT AT(14,24)VALIDATE
("JN")SIZE(-1):J$ :: CALL CL
EAR :: IF J$="J" THEN OPEN #
3:"PIO"
190 PRINT " AUSGABE ANHALTEN
 >SPACE<":;:" AUSGABE ABBREC
HEN >BACK<":;:
200 CALL PEEK(ANF,A,B,C,D,E,
F,G,H):: CALL DH2(A,A$):: CA
LL DH2(B,B$):: CALL DH2(C,C$
):: CALL DH2(D,D$):: CALL DH
2(E,E$):: CALL DH2(F,F$)
210 CALL DH2(G,G$):: CALL DH
2(H,H$):: CALL DH(ANF,AD$)::
 PRINT AD$;" ";A$;" ";B$;" "
;C$;" ";D$;" ";E$;" ";F$;" "
;G$;" ";H$ :: IF J$<>"J" THE
N 230
220 PRINT #3:"      ";AD$;"
 ";A$;"    ";B$;"    ";C$;"
";D$;"    ";E$;"    ";F$;"    "
;G$;"    ";H$
230 CALL KEY(Ø,T,SE):: IF T=
15 THEN 250 ELSE IF SE<>Ø TH
EN 230
240 ANF=ANF+8 :: IF ANF<ADE
THEN 200
250 IF J$="J" THEN CLOSE #3
260 PRINT :: INPUT " ABSPEIC
HERN >J/N< ":SP$ :: IF SP$<>
"J" THEN 110
270 DISPLAY AT(20,5)ERASE AL
L:"DISK ? UND DATEI ?": :"
  >DSK1.           <" :: ACCE
PT AT(22,9)VALIDATE("12")SIZ
E(-1):DI$
280 ACCEPT AT(22,11)VALIDATE
(UALPHA,DIGIT)SIZE(-10):DT$
:: DSK$="DSK"&DI$&"."&DT$
290 OPEN #4:DSK$,OUTPUT,RELA
TIVE,INTERNAL :: RE=Ø
300 FOR Z=ADA TO ADE STEP 8
:: CALL PEEK(Z,A1,B1,C1,D1,E
1,F1,G1,H1):: PRINT #4,REC R
E:A1,B1,C1,D1,E1,F1,G1,H1
310 RE=RE+1 :: NEXT Z :: CLO
SE #4 :: GOTO 110
320 DISPLAY AT(2,4)ERASE ALL
:"RAM-AENDERUNG (4erBLOCK)":
 : :"   eingabe in hex   ___
_": : :" ram start address >
    <"
330 ACCEPT AT(8,21)VALIDATE(
"Ø123456789ABCDEF")SIZE(4):R
A$ :: CALL HD(RA,RA$):: IF R
A>32767 THEN RA=-65536+RA
340 CALL DH(RA,RA$):: DISPLA
Y AT(10,15):"dez> ";RA :: CA
LL PEEK(RA,T,U,V,W):: CALL D
H2(T,T$):: CALL DH2(U,U$)::
CALL DH2(V,V$):: CALL DH2(W,
W$)
350 Q$="  " :: DISPLAY AT(14
,4):RA$&Q$&T$&Q$&U$&Q$&V$&Q$
&W$
360 ACCEPT AT(14,10)VALIDATE
```

```
("Ø123456789ABCDEF")SIZE(-2)
:T1$
370 ACCEPT AT(14,14)VALIDATE
("Ø123456789ABCDEF")SIZE(-2)
:U1$
380 ACCEPT AT(14,18)VALIDATE
("Ø123456789ABCDEF")SIZE(-2)
:V1$
390 ACCEPT AT(14,22)VALIDATE
("Ø123456789ABCDEF")SIZE(-2)
:W1$
400 CALL HD(T1,T1$):: CALL H
D(U1,U1$):: CALL HD(V1,V1$):
: CALL HD(W1,W1$):: CALL LOA
D(RA,T1,U1,V1,W1)
410 DISPLAY AT(22,3)BEEP:"WE
ITER SCHREIBEN > W <": :"
   ZUM MENUE > REDO <"
420 CALL KEY(Ø,KE,SI):: IF S
I=Ø THEN 420 ELSE IF KE=6 TH
EN 110 ELSE IF KE=87 THEN 43
Ø ELSE 420
430 DISPLAY AT(22,2):"": :""
 :: RA=RA+4 :: GOTO 340
440 DISPLAY AT(10,2)ERASE AL
L BEEP:"BITTE DEN SPEICHERBE
REICH,": :"  DER BELEGT WERD
EN SOLL": :"   EINGEBEN!"
450 DISPLAY AT(14,15)BEEP:">
 SPACE <" :: CALL KEY(Ø,O,P)
:: DISPLAY AT(14,15):"" :: I
F P=Ø THEN 450
460 CALL ADD(BDA,BDE):: DISP
LAY AT(2Ø,5):"DISK ? UND DAT
EI ?": :"     >DSK1.
 <" :: ACCEPT AT(22,9)VALIDA
TE("12")SIZE(-1):DI$
470 ACCEPT AT(22,11)VALIDATE
(UALPHA,DIGIT)SIZE(-1Ø):DT$
:: DSK$="DSK"&DI$&"."&DT$
480 OPEN #4:DSK$,INPUT ,RELA
TIVE,INTERNAL :: RE=Ø
490 FOR Z=BDA TO BDE STEP 8
:: INPUT #4:A1,B1,C1,D1,E1,F
1,G1,H1 :: CALL LOAD(Z,Ai,B1
,C1,D1,E1,F1,G1,H1)
500 IF EOF(4)THEN 520 ELSE R
E=RE+1
510 NEXT Z
520 CALL DH(BDA,BDA$):: CALL
 DH(Z+8,BDE$):: DISPLAY AT(1
Ø,5)ERASE ALL:"DATEI-INHALT
WURDE": :"  IN ADDRESSE "&BD
A$&"-"&BDE$: :"         ABGE
LEGT" :: CLOSE #4
530 DISPLAY AT(23,7)BEEP:"ZU
M MENUE > REDO <" :: CALL KE
Y(Ø,KE,SI):: IF SI=Ø THEN 53
Ø ELSE IF KE=6 THEN 110 ELSE
 530
540 SUB ADD(ADA,ADE):: DISPL
AY AT(2,4)ERASE ALL:"eingabe
 in hex    ____": :" ram star
t address >    <": : : :" ra
m last  address >    <"
550 ACCEPT AT(4,21)VALIDATE(
"Ø123456789ABCDEF")SIZE(-4):
ADA$ :: IF ADA$="" THEN 550
ELSE CALL HD(ADA,ADA$)
560 ACCEPT AT(8,21)VALIDATE(
"Ø123456789ABCDEF")SIZE(-4):
ADE$ :: IF ADE$="" THEN 550
ELSE CALL HD(ADE,ADE$)
570 IF ADA>32767 THEN ADA=-6
5536+ADA
580 IF ADE>32767 THEN ADE=-6
5536+ADE
590 DISPLAY AT(6,1Ø):"dez>";
ADA :: DISPLAY AT(1Ø,1Ø):"de
z>";ADE :: SUBEND
600 SUB HD(D,H$):: F(1)=4096
 :: F(2)=256 :: F(3)=16 :: F
(4)=1 :: HZ$="Ø123456789ABCD
EF" :: D=Ø :: H$="ØØØØ"&H$
610 H$=SEG$(H$,LEN(H$)-3,4):
: FOR N=1 TO 4 :: D=(POS(HZ$
,SEG$(H$,N,1),1)-1)*F(N)+D :
: NEXT N :: SUBEND
620 SUB DH(D,H$):: HZ$="Ø123
456789ABCDEF" :: H$,H1$="" :
: D1=D
630 FOR N=1 TO 4 :: R=(D1-IN
T(D1/16)*16):: D1=INT(D1/16)
:: H1$=H1$&SEG$(HZ$,R+1,1)::
 NEXT N
640 FOR N=1 TO 4 :: H$=H$&SE
G$(H1$,5-N,1):: NEXT N :: SU
BEND
650 SUB DH2(D,H$):: HZ$="Ø12
3456789ABCDEF" :: H$,H1$=""
:: D1=D
660 FOR N=1 TO 4 :: R=(D1-IN
T(D1/16)*16):: D1=INT(D1/16)
:: H1$=H1$&SEG$(HZ$,R+1,1)::
 NEXT N
670 FOR N=3 TO 4 :: H$=H$&SE
G$(H1$,5-N,1):: NEXT N :: SU
BEND
```

LISTINGS

STARWARS: DAS IMPERIUM SCHLÄGT ZURÜCK!

Das Spiel für alle Star-Wars-Freunde und zukünftige Jedi-Ritter!!! Sie sind Luke Skywalker. Sie besteigen Ihren X-Wing-Fighter und fliegen zum Todesstern. Zerstören Sie diese gewaltige Waffe des bösen galaktischen Imperiums. Han Solo gibt Ihnen mit seinem superschnellen Melinium Falcon Rükkendeckung. Aber Tie-Fighter (rot) greifen Sie an!!! Manövrieren Sie geschickt und setzen Sie Ihren Laser ein. Wenn Lord Darth Vader, der böse Vertreter der dunklen Seite der Macht, auftaucht, müssen Sie schnell reagieren!
Nur dann ist die Macht mit Ihnen und Sie können in den Todesstern eindringen. Nun ist Spannung und Nervenkitzel garantiert:
Sie können nichts anderes tun, als den Schüssen des Tie-Fighters auszuweichen!! Aber Vorsicht!!! Rammen Sie dabei nicht die Wand des Todessterns!! Wenn der kleine Entlüftungsschacht auftaucht, betätigen Sie den Laser! Ja, die Macht ist mit Ihnen: Sie haben den schrecklichen Todesstern zerstört!!!
Ich verspreche Ihnen viel Spannung und Spaß bei diesem fantastischem Märchen in einer weit, weit entfernten Galaxis. Es war einmal in ferner Zukunft . . .
Die galaktischen Ausmaße des fantastischen Märchens „Krieg der Sterne" wurden durch eine wirkliche detailgenaue, ausgetüftelte Programmiermethode erreicht. Das Programm enthält eine Reihe Besonderheiten, die man beim TI99 nicht für möglich gehalten hätte: Von Riesenbuchstaben beim Titelbild bis hin zur quasi 3D Grafik in Screen 2.
Eine gleichbleibende diagonale Bewegung der Sprites, wie in Screen 2, ist eigentlich hardwarebedingt unmöglich!!!
Ein diagonales Scrollen war aber unerläßlich (wie z.B. in „Zaxxon"). So habe ich also unsichtbare Sprites rechts unten und links oben definiert.
Diese verhindern, daß man die sich schräg bewegenden Sprites dort sieht! Natürlich kann man das Programm problemlos abändern:



Mit sehr ansprechender Grafik: Starwars

Das Spiel wird z.B. schwerer, wenn man die Zahl hinter "RND<." in Zeile 760 verkleinert.
Durch jeweils umgekehrte Änderung läßt sich das Spiel erleichtern.
Gesteuert wird mit Joystick 1 (Screen 1: rauf–runter; Screen 2: links–rechts).
Laser: Fire-Taste
Alpha Lock bitte entrasten.

Programmaufbau

100–160	Vorspann (Anleitung Star Wars I)
170–220	Titelbild
230–310	Definitionen/ Aufbau von Screen 1
320–510	Angriff der Tie-Fighter
520–560	Angriff von Darth Vader
570–670	Definitionen/ Aufbau von Screen 2
680–930	Schlacht im Todesstern
940–980	Unterprogramme

Variablen-Liste

LA	Laser
Fa	Farbe
P	Position des X-Wing-Fighters
SP()	Tie-Fighter (Sprites 3–6)
PU	Punkte
RUI	Runde
I1	Schnelligkeit der Tie-Fighter
SCH	verbrauchte Schiffe (max. 3)
I,Z	allgemein benutzt (in For...Next, etc.)
Z1,Z2	allgemein benutzt, in CALL JOST/ Key
Z$	allgemein benutzt, zum Lesen der CHARs in DATAs

Wichtig!!!

„Krieg der Sterne" kann man auch ohne den zweiten Teil, „Das Imperium schlägt zurück" spielen. Dann muß aber die Zeile 920 folgendermaßen abgeändert werden: "REDO 8" anstatt "WEITER 8" und RUN 240 anstatt RUN "CS1".
Man kann „Das Imperium schlägt zurück" auch ohne vorhergehendes Meistern von „Krieg der Sterne" spielen. In dem Fall kann man im Imperium-Programm die Zeile 245 und des CALL CHAR in Zeile 250 ersatzlos löschen.
Wenn man jedoch, wie eigentlich vorgesehen, beide Spiele haben möchte,

Bitte lesen Sie weiter auf Seite 19

LISTINGS

```
10 ! *********************
11 ! *      STARWAR1      *
12 ! *  Krieg der Sterne  *
13 ! *                    *
14 ! *    Copyright by    *
15 ! *                    *
16 ! *  Patrick  Schmitz  *
17 ! *                    *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole  *
21 ! *     Ext. Basic     *
22 ! *     Joystick 1     *
23 ! *    Cassettenrec.   *
24 ! *(oder Disk+32K-Erw.)*
25 ! *                    *
26 ! *  Speicherbelegung  *
27 ! *    12272 Bytes     *
28 ! *                    *
29 ! *********************
100 RANDOMIZE :: CALL CLEAR
:: CALL SCREEN(2):: FOR Z=1
TO 12 :: CALL COLOR(Z,11,1):
: NEXT Z :: CALL MAGNIFY(3):
: CALL STERNE
110 DISPLAY AT(1,1):"Luke,Du
 musst den Todessterndes boe
sen galaktischen":"Imperiums
 zerstoeren.":"Zuerst bist D
u mit Deinem X-Wing Fighter
ueber der Ober-flaeche"
120 DISPLAY AT(6,9):"des Tod
essterns, wo":"Du angreifend
e Tie-Fighter..(rot) zerstoe
ren musst. Wennder Darth Vad
er-Tie-Fighter (gruen) aufta
ucht, musst Du"
130 DISPLAY AT(11,1):"im ric
htigen Moment reagie- ren, d
ann ist die Macht mit Dir. D
u befindest Dich nun..im Tod
esstern. Weiche den...Schues
sen aus und treffe denLuf"
140 DISPLAY AT(16,4):"schach
t.": :"Treffer: 100 Punkte":
"Ausweichen: 20 Punkte":"Max
. Laser: 20":"Max. Schiffe:
3":"Bonus pro uebr.Schiff 10
0 P.": :">ENTER" :: ACCEPT V
ALIDATE(""):Z$
150 DATA 262.4,392.4,349.2,3
30.2,294.2,523.4,392.3,349.2
,330.2,294.2,523.4,392.3,349
.2,330.2,349.2,294.4
160 CALL CLEAR :: FOR Z=65 T
O 90 :: IF Z=66 OR Z=74 OR Z
=79 OR Z=81 OR Z=85 OR Z=86
OR Z=89 THEN 180
170 CALL CHARPAT(Z,Z$):: FOR
 Z1=1 TO 15 STEP 2 :: Z1$=Z1
$&RPT$(SEG$(Z$,Z1,2),2):: NE
XT Z1 :: CALL CHAR(Z,SEG$(Z1
$,1,16),Z+32,SEG$(Z1$,17,16)
):: Z1$=""
180 NEXT Z :: DISPLAY AT(1,1
):"ES WAR EINMAL":"es war ei
nmal":"   IN EINER WEIT":"..
.in einer weit,":"WEIT ENTFE
RNTEN GALAXIS":"weit entfern
ten galaxis..."
190 DISPLAY AT(11,11):"KRIEG
":TAB(11);"krieg":TAB(12);"D
ER":TAB(12);"der":TAB(8);"S
T E R N E":TAB(8);"s t e r n
 e" :: CALL CHAR(40,"3C4299A
1A199423C")
200 DISPLAY AT(21,10):"( 198
4": :"....PATRICK  SCHMITZ":
"....patrick  schmitz" :: CA
LL STARWARS(27):: CALL HCHAR
(1,1,32,196):: CALL HCHAR(21
,1,32,128)
210 DISPLAY AT(1,11):"*STAR*
":TAB(10);"**star**":TAB(9);
"***WARS***":TAB(8);"****war
s****" :: DISPLAY AT(22,4):"
DIE MACHT SEI MIT DIR":"...d
ie macht sei mit dir."
220 CALL STARWARS(5):: CALL
CLEAR :: CALL DELSPRITE(ALL)
:: CALL CHARSET :: CALL STER
NE :: DISPLAY AT(1,1):"KRIEG
 DER STERNE * STAR WARS":"LA
SER: 20";TAB(17);"PUNKTE:
0"
230 DATA 00000F060603C37FFFC
30306060F00000000E000000000F
EFF00000000E00000,0000207070
7173FFFF7371707020000000000004
0E0E8ECEFFFFCE8E0E0E040000
240 DATA 0E1C3870E0E1E7FFFFE
7E1E070381C0E70381C0E0787E7F
FFFE787070E1C38F0,0002070C0C
06030103030 70F1F3F79F0000000
0C1E1E3CF0E0E0B018188EC2E0
250 DATA 000001073F3F3B070E1
C3838787DEFE30C78F0E08101030
303133B7FFFFFFFFF,807C707C5F
5F070701011138393B7BFF000000
008080F0F0F0F0FEFEFCFCF8F8
260 DATA 1F1F303030306161636
7C7CFDEDED6E4F8F80C1C2CECCCC
CC6C6C3E36363CCCC
270 RESTORE 230 :: FOR Z=96
TO 120 STEP 4 :: READ Z$ ::
CALL CHAR(Z,Z$):: NEXT Z
280 CALL CHAR(143,RPT$("F",1
6)):: CALL HCHAR(21,1,143,12
8):: FOR Z=1 TO 4 :: CALL SP
RITE(#10+Z,104+Z*4,2,145,63*
Z,0,-20):: NEXT Z
290 CALL CHAR(136,"0000FF"&R
PT$("0",20)&"FF"&RPT$("0",36
)):: CALL SPRITE(#10,136,9,2
30,230,0,110):: CALL CHAR(12
4,RPT$("0900049104200 0A0",4)
):: LA=20 :: P=80
```

```
300 CALL CHAR(128,RPT$("0800
800100010040",4)):: CALL XWI
NG(2,13):: CALL XWING(2,16):
: CALL CHAR(135,RPT$("0",14)
&"FF"):: CALL SPRITE(#1,96,1
2,80,16):: FA=7
310 FOR Z=11 TO 14 :: CALL C
OLOR(#Z,5):: NEXT Z :: CALL
COLOR(9,11,2,13,9,2,14,5,2):
: FOR Z=1 TO 8 :: CALL COLOR
(Z,12,2):: NEXT Z :: FOR Z=3
 TO 6 :: SP(Z)=-1 :: NEXT Z
:: I=0
320 FOR Z=3 TO 6 :: IF NOT S
P(Z)THEN 350 ELSE CALL COINC
(#1,#Z,8,Z1):: IF NOT Z1 THE
N 350 ELSE CALL DELSPRITE(#Z
):: SP(Z)=0
330 FOR Z=1 TO 2 :: CALL PAT
TERN(#1,124):: CALL SOUND(-4
00,-6,0,110,5):: CALL PATTER
N(#1,128):: CALL SOUND(-400,
-7,0,120,5):: NEXT Z :: GOTO
 380
340 PU=0 :: DISPLAY AT(2,24)
:"...0" :: CALL COLOR(#1,12)
:: IF SP(3)+SP(4)+SP(5)+SP(6
)<>0 THEN 320 ELSE 490
350 NEXT Z :: CALL JOYST(1,Z
1,Z2):: IF Z2 AND P-Z2*2>24
THEN 360 ELSE CALL KEY(1,Z1,
Z2):: IF Z1<>18 THEN IF RND>
.5 THEN 470 ELSE 320 ELSE 37
0
360 IF P-Z2*2>128 THEN P=P-8
 :: GOTO 330 ELSE P=P-Z2*2 :
: CALL LOCATE(#1,P,16):: CAL
L KEY(1,Z1,Z2):: IF Z1<>18 T
HEN 320
370 IF LA>0 THEN 410 ELSE CA
LL MOTION(#1,-20,80):: CALL
SOUND(-100,440,2,442,8,438,8
):: CALL SOUND(-100,440,6,44
2,10,438,10):: CALL MOTION(#
1,0,0)
380 CALL COLOR(#1,2):: IF SC
H<2 THEN SCH=SCH+1 :: CALL P
ATTERN(#1,96):: GOTO 400 ELS
E CALL DELSPRITE(#3,#4,#5,#6
):: CALL SO
390 DISPLAY AT(5,1):"LUKE, E
S IST DIR NOCH NICHT": :"GEL
UNGEN, DEN TODESSTERN ZU": :
"ZERSTOEREN. ABER " :: CALL
ENDE :: CALL SOUND(-2,1400,0
):: RUN 220
400 CALL HCHAR(2,3*SCH+10,32
,2):: CALL HCHAR(3,3*SCH+10,
32,2):: P=80 :: CALL LOCATE(
#1,P,16):: LA=20 :: DISPLAY
AT(2,7)SIZE(3):LA :: GOTO 34
0
410 LA=LA-1 :: DISPLAY AT(2,
7)SIZE(3):LA :: FOR Z=3 TO 6
 :: IF NOT SP(Z)THEN 420 ELS
E CALL POSITION(#Z,Z1,Z2)::
IF INT(Z1/8)=INT(P/8)THEN 44
0
420 NEXT Z :: CALL SOUND(-10
0,900,4,902,8,898,8,-4,8)::
CALL HCHAR((P+8)/8,4,135,29)
:: CALL SOUND(-100,880,4,882
,8,878,8,-4,12)
430 CALL HCHAR((P+8)/8,4,32,
29):: GOTO 320
440 CALL DISTANCE(#Z,P,24,Z1
):: CALL LOCATE(#10,230,1)::
 IF SQR(Z1)<17 THEN 320 ELSE
 CALL SOUND(-100,900,4,902,8
):: CALL HCHAR((P+8)/8,4,135
,SQR(Z1)/8-1)
450 CALL HCHAR((P+8)/8,4,32,
29):: CALL PATTERN(#Z,124)::
 CALL SOUND(-100,120,5,130,8
,1400,30,-8,3):: CALL PATTER
N(#Z,128)
460 CALL SOUND(-100,110,5,12
0,8,2000,30,-8,3):: PU=PU+10
0 :: DISPLAY AT(2,24):PU ::
SP(Z)=0 :: CALL DELSPRITE(#Z
):: IF SP(3)+SP(4)+SP(5)+SP(
6)<>0 THEN 320 ELSE 490
470 IF I=4 THEN 320 ELSE I=I
+1 :: CALL SPRITE(#I+2,100,F
A,1,256):: CALL SOUND(-100,-
2,2,110,3)
480 FOR Z=5 TO 10 :: CALL MO
TION(#I+2,Z+INT(RND*29)+7,-Z
*2):: NEXT Z :: CALL POSITIO
N(#I+2,Z,Z1):: CALL MOTION(#
I+2,0,-20):: CALL LOCATE(#I+
2,INT(Z/16)*16,Z1-6):: GOTO
320
490 RU1=RU1+1 :: IF RU1>=2 T
HEN 500 ELSE FA=14 :: I1=1 :
: GOTO 310
500 CALL SPRITE(#2,104,4,1,1
92):: FOR Z=10 TO 0 STEP -10
 :: CALL SOUND(-100,110+Z,2,
120+Z,4,130+Z,6):: NEXT Z
510 CALL MOTION(#2,24,-16)::
 CALL POSITION(#2,Z,Z2):: CA
LL KEY(1,Z1,Z2):: IF Z1=18 T
HEN IF INT((Z+4)/32)=INT(P/3
2)THEN CALL MOTION(#2,-1,-3)
:: GOTO 540 ELSE 520 ELSE IF
 Z<112 THEN 510 ELSE 530
520 CALL SPRITE(#10,136,9,P,
24,0,100):: FOR Z=1 TO 2 ::
CALL SOUND(-10,600,4,602,8):
: CALL SOUND(-10,580,4,578,8
):: NEXT Z :: CALL DELSPRITE
(#10)
530 CALL MOTION(#2,1,-125)::
 CALL SOUND(-100,1040,5,1060
,6,-3,7):: CALL SOUND(-100,1
```

LISTINGS

```
030,6,1040,8,-2,10):: CALL D
ELSPRITE(#2):: FA=7 :: GOTO
310
540 CALL SPRITE(#10,136,9,P,
24,0,70):: FOR Z=1 TO 2 :: C
ALL SOUND(-10,600,4,602,8,59
8,8):: CALL SOUND(-10,580,4,
578,8,582,8):: NEXT Z
550 CALL MOTION(#2,0,0):: CA
LL DELSPRITE(#10):: FOR Z=1
TO 2 :: CALL SOUND(-100,110,
30,110,30,Z*1000+250,30,-8,2
):: CALL PATTERN(#2,120+Z*4)
:: NEXT Z
560 CALL DELSPRITE(#2):: DIS
PLAY AT(4,1):"DIE..M A C H T
..IST MIT DIR!" :: CALL STAR
WARS(10):: CALL DELSPRITE(AL
L):: CALL CLEAR :: CALL COLO
R(13,2,2,14,2,2):: P=2
570 CALL CHAR(134,RPT$("F",1
6)):: CALL CHAR(141,"0103070
F1F3F7FFFFFFEFCF8F0E0C080"):
: CALL CHAR(96,"00000408103F
7F8103070E1C38F0202000060E1C
38F0E0E0E060646870604080")
580 CALL CHAR(108,"C3F1797D3
F9FCF7F7B707078FFFF070F0080C
0C0C2868ECEE6F37BFFFFDF8F07"
):: CALL CHAR(112,"01FCFE030
7FCF90606FDF80503FEFC08")
590 CALL CHAR(114,"387C0C068
3C3663E3E6ECE9E1E3F3F7F")::
CALL CHAR(116,"8143261C1C366
3CD9320FFFFFF000000F0F038381
C1C0E8EC763FEFEFE3E7FC3")
600 CALL CHAR(120,"40BC7F3F1
C1E1F3F6F7F9F3333234100060F
0F060C0F0FCFFE3F1B89C8EC6E1"
):: CALL CHAR(136,"000000000
000000000002183870E0C0"&RPT
$("0",32))
610 FOR Z=1 TO 24 :: CALL HC
HAR(Z,26-Z,143,7):: CALL HCH
AR(Z,25-Z,141):: CALL HCHAR(
Z,33-Z,142):: NEXT Z :: FOR
Z=2 TO 8 :: CALL VCHAR(Z,34-
Z,134,Z+4):: NEXT Z
620 FOR Z=19 TO 24 :: CALL H
CHAR(Z,34-Z,134,31-Z):: NEXT
 Z :: FOR Z=8 TO 18 :: CALL
HCHAR(Z,34-Z,134,Z-7):: NEXT
 Z :: FOR Z=1 TO 6 :: CALL H
CHAR(Z,-2*Z+20,134,Z+5)
630 CALL VCHAR(-2*Z+19,Z,134
,Z+6):: NEXT Z :: FOR Z=7 TO
 17 :: CALL HCHAR(Z,7,134,18
-Z):: NEXT Z :: CALL COLOR(1
3,8,2,14,5,8)
640 CALL SPRITE(#14,116,8,1,
189,14,-16,#1,96,12,100,106,
#15,108,8,1,250,14,-16):: FO
R Z=2 TO 5 :: CALL SPRITE(#Z
,140,1,1,1)
650 CALL SPRITE(#Z+4,140,1,1
78,240,#Z+8,140,1,172,240)::
 NEXT Z :: FOR Z=SCH*3+26 TO
 29 STEP 3 :: CALL XWING(23,
Z):: NEXT Z :: DISPLAY AT(22
,22):"SCHIFFE"
660 DISPLAY AT(1,1)SIZE(9):"
STAR WARS" :: DISPLAY AT(3,1
)SIZE(6):"PUNKTE" :: CALL W
:: CALL SPRITE(#16,120,8,1,1
89,14,-16,#17,112,8,1,250,14
,-16)
670 FOR Z=20 TO 24 :: CALL S
PRITE(#Z,36,8,INT(RND*17)+11
2,INT(RND*11)+246,14,-16,#Z+
4,36,8,INT(RND*20)+70,INT(RN
D*15)+1,14,-16):: NEXT Z ::
DISPLAY AT(4,1)SIZE(4):PU
680 CALL SOUND(-4250,-7,10):
: CALL JOYST(1,Z1,Z2):: IF Z
1=-4 THEN IF P=1 THEN P=0 EL
SE IF P=2 THEN P=1 ELSE IF P
=3 THEN P=2 ELSE 680 ELSE IF
 Z1=4 THEN IF P=3 THEN P=4.5
 ELSE IF P=2 THEN P=3 ELSE I
F P=1 THEN P=2 ELSE 680
690 CALL LOCATE(#1,4*P+92,4*
P+96):: IF P=0 THEN 700 ELSE
 IF P=4.5 THEN 710 ELSE IF R
ND<.040 THEN 790 ELSE IF RND
>.5 THEN 680 ELSE 760
700 CALL COINC(#1,#14,8,Z2):
: IF Z2 THEN 720 ELSE CALL C
OINC(#1,#16,8,Z2):: IF Z2 TH
EN 720 ELSE CALL JOYST(1,Z1,
Z2):: IF Z1=4 THEN CALL LOCA
TE(#1,96,100):: P=1 :: GOTO
680 ELSE 700
710 CALL COINC(#1,#15,8,Z2):
: IF Z2 THEN 720 ELSE CALL C
OINC(#1,#17,8,Z2):: IF Z2 TH
EN 720 ELSE CALL JOYST(1,Z1,
Z2):: IF Z1=-4 THEN CALL LOC
ATE(#1,104,108):: P=3 :: GOT
O 680 ELSE 710
720 FOR Z=1 TO 2 :: CALL PAT
TERN(#1,120+Z*4):: CALL SOUN
D(-100,110,25,110,25,1000+Z*
500,25,-8,2):: NEXT Z :: SCH
=SCH+1
730 CALL HCHAR(23,SCH*3+23,3
2,2):: CALL HCHAR(24,SCH*3+2
3,32,2):: PU=0 :: DISPLAY AT
(4,1)SIZE(5):PU :: IF SCH<=2
 THEN P=2 :: CALL PATTERN(#1
,96):: GOTO 680
740 CALL SO :: CALL DELSPRIT
E(ALL):: CALL CLEAR :: DISPL
AY AT(3,1):"LUKE, DU BIST ZW
AR IN DEN"
750 DISPLAY AT(5,1):"TODESST
ERN EINGEDRUNGEN, DU": :"KON
```

```
NTEST IHN ABER NOCH NICHT":
:"ZERSTOEREN. ABER" :: CALL
ENDE :: CALL SOUND(-2,1400,0
):: RUN 220
760 IF I=1 THEN 770 ELSE I=1
 :: CALL SOUND(-100,-3,5,220
,3,224,6):: Z3=INT(RND*3)+1
:: CALL SPRITE(#19,100,7,Z3*
4+136,Z3*4+48):: GOTO 680
770 I=0 :: CALL SPRITE(#18,1
36,4,Z3*4+136,Z3*4+48,-28,32
):: FOR Z=-1 TO -3 STEP -1 :
: CALL SOUND(-100,Z,4,600+Z*
10,8,800+Z*5,9):: NEXT Z
780 IF P=Z3 THEN CALL DELSPR
ITE(#18):: GOTO 720 ELSE CAL
L SOUND(-100,110,4):: CALL S
OUND(-100,115,8):: CALL DELS
PRITE(#18):: PU=PU+20 :: DIS
PLAY AT(4,1)SIZE(4):PU :: GO
TO 680
790 CALL CHAR(44,RPT$("0",13
)&"10001"&RPT$("0",26)&"4080
4"):: CALL SPRITE(#20,44,2,1
,216,14,-16):: CALL SOUND(-1
00,1000,4,1010,6):: IF P=2 T
HEN 830
800 CALL POSITION(#20,Z1,Z2)
:: IF Z1>88 THEN 820 ELSE CA
LL JOYST(1,Z1,Z2):: IF Z1=-4
 AND P=3 OR Z1=4 AND P=1 THE
N 830
810 CALL KEY(1,Z1,Z2):: IF Z
1<>18 THEN 800 ELSE CALL SPR
ITE(#18,136,10,4*P+92,4*P+96
,-28,32):: CALL SOUND(-100,6
60,4,662,7):: CALL SOUND(-10
0,110,5)
820 FOR Z=900 TO 850 STEP -1
0 :: CALL SOUND(-100,Z,4,Z+3
,10,Z-3,10):: NEXT Z :: I=0
:: CALL DELSPRITE(#18,#20)::
 GOTO 680
830 CALL LOCATE(#1,100,104):
: CALL KEY(1,Z1,Z2):: IF Z1=
18 THEN PU=PU+(3-SCH)*100 ::
 GOTO 840 ELSE CALL POSITION
(#20,Z1,Z2):: IF Z1>88 THEN
820 ELSE 830
840 CALL MOTION(#20,0,0):: C
ALL SPRITE(#18,136,10,4*0+92
,4*P+96,-28,32):: CALL SOUND
(-10,600,4,602,8):: CALL SOU
ND(-10,578,8)
850 CALL DELSPRITE(#18,#19):
: FOR Z=1 TO 16 :: CALL COLO
R(#20,Z):: NEXT Z :: CALL MO
TION(#1,-5,-10):: FOR Z=1E3
TO 1200 STEP 50 :: CALL SOUN
D(-100,Z,4):: NEXT Z :: CALL
 LOCATE(#1,44,20)
860 CALL MOTION(#1,0,0):: CA
LL COLOR(#20,16):: CALL SOUN
D(-100,-5,3):: CALL PATTERN(
#20,124):: CALL SOUND(-100,-
6,3):: CALL PATTERN(#20,124)
:: FOR Z2=1 TO 2
870 CALL CHAR(134,"000100800
40030"):: CALL CHAR(141,"008
0010400300701008002020110503
0010080302005001"):: CALL PA
TTERN(#14,124,#15,128,#16,12
4,#17,128)
880 CALL COLOR(13,7,2,14,7,2
):: FOR Z=14 TO 17 :: CALL C
OLOR(#Z,7):: NEXT Z :: FOR Z
=1 TO 3 :: CALL SOUND(-4250,
110,30,110,30,1E3+Z*400,30,-
8,0):: NEXT Z
890 CALL CHAR(134,"000010008
001"):: CALL CHAR(141,"00010
0002005000040000020080000100
080080030"):: CALL COLOR(1,7
,7,13,12,2,14,12,2,1,11,2)
900 CALL SOUND(-999,-6,0)::
NEXT Z2 :: CALL DELSPRITE(AL
L):: CALL CLEAR :: CALL STER
NE :: DISPLAY AT(1,1):"LUKE.
 ES IST DIR GELUNGEN": :"DEN
 TODESSTERN ZU": :"ZERSTOERE
N !!!"
910 DISPLAY AT(8,1):"HAN. LE
IA. CHEWI. R2 UND 3PO": :"FR
EUEN SICH. OBI WAN UND": :"J
ODA SIND STOLZ AUF DICH !":
: : :"DIE M A C H T IST MIT
DIR !!": :PU;" PUNKTE"
920 CALL STARWARS(10):: DISP
LAY AT(24,1):"WEITER>8  ENDE
>4" :: FOR Z=1 TO 200 :: DIS
PLAY AT(24,24):200-Z :: CALL
 KEY(3,Z1,Z2):: IF Z1=56 THE
N CALL CLEAR :: RUN "CS1" EL
SE IF Z1=52 THEN Z=300
930 NEXT Z :: IF Z<300 THEN
RUN 220 ELSE CALL DELSPRITE(
ALL):: CALL CHARSET :: PRINT
 : : : : :"DAS I M P E R I U
 M WIRD": :"ZURUECKSCHLAGEN
UND": :"DIE J E D I RITTER W
ERDEN": :"ZURUECKKEHREN.....
.": : : :
940 SUB XWING(Z1,Z2):: FOR Z
=96 TO 97 :: CALL HCHAR(Z1+Z
-96,Z2,Z):: CALL HCHAR(Z1+Z-
96,Z2+1,Z+2):: NEXT Z :: SUB
END :: SUB SO :: FOR Z=1 TO
3 :: FOR Z1=0 TO 9 STEP 3 ::
 CALL SOUND(-100,110,Z1):: N
EXT Z1 :: NEXT Z
950 SUBEND :: SUB ENDE :: DI
SPLAY AT(11,1):"DER KRIEG DE
R  S T E R N E": :"GEHT WEIT
ER...";TAB(23);"4>ENDE": :TA
B(23);"8>REDO" :: FOR Z=1 TO
 200 :: CALL KEY(3,Z1,Z2)::
```

```
IF Z1=56 OR Z1=6 THEN SUBEXI
T
960 IF Z1=52 OR Z1=2 THEN CA
LL DELSPRITE(ALL):: CALL CLE
AR :: CALL CHARSET :: PRINT
"BIS ZUM NAECHSTEN MAL !": :
 :"DIE M A C H T SEI MIT DIR
 !": : : : : : :: END ELSE DIS
PLAY AT(16,11):200-Z
970 NEXT Z :: SUBEND :: SUB
STERNE :: CALL CHAR(36,"18"&
RPT$("0",62)):: FOR Z=18 TO
28 :: CALL SPRITE(#Z,36,16,R
ND*136+16,252,0,RND*-16-5)::
 NEXT Z :: SUBEND :: SUB W :
: CALL SOUND(1200,110,8)
980 CALL SOUND(2,112,4):: SU
BEND :: SUB STARWARS(Z2):: R
ESTORE 150 :: FOR Z=1 TO 16
:: READ Z1 :: CALL SOUND((Z1
-INT(Z1))*1200,Z1,5,Z1+2,Z2,
Z1-2,Z2):: NEXT Z :: SUBEND
```

LISTINGS

STARWARS

Fortsetzung von Seite 14

gibt man zunächst das „Krieg der Sterne"-Listing ein. Dann speichert man das Programm ab. Danach gibt man „Das Imperium schlägt zurück" ein und speichert es auf der gleichen Kassette einige Sekunden hinter dem 1. Teil ab. Zum Spielen lädt man zuerst „Krieg der Sterne". Wenn man den Todesstern zerstört hat, wird „Imperium" automatisch geladen (Bildschirmanweisungen* befolgen!). (*"Rewind Cassette Type" ist natürlich nicht nötig, da das "Imperium"-Programm ja gleich dem bereits geladenem „Krieg der Sterne"-Programm folgt.)

DAS IMPERIUM SCHLÄGT ZURÜCK

Es war einmal in ferner Zukunft . . .
In einer weit, weit entfernten Galaxis brach der Krieg der Sterne aus.
Den Rebellen war es gelungen, den Todesstern zu zerstören. Doch Darth Vader, der Vertreter der bösen Seite der Macht, war entkommen.
Er verfolgt nun gnadenlos die Allianz der Rebellen. Diese haben sich in die Schneewüste des Eisplaneten Hoth zurückgezogen. Ihre Basis wird von AT-ATs (= All Terrain Armoured Transport), riesigen mechanischen Monstern des Imperiums, angegriffen. Der Spieler schlüpft nun in die Rolle von Luke Skywalker.
Die Walkers (= AT-ATs) müssen aufgehalten werden, indem man sie mit Hilfe eines Snowspeeders (blau) zerstört!!! Aber dazu sind 10 (zehn!) Treffer notwendig! Trifft man ein sich bewegendes AT-AT, bleibt dieses meist stehen. Aber manchmal bewegt es sich auch mit erhöhter Geschwindigkeit weiter! Trifft man ein stehendes AT-AT, fängt dieses an, sich zu bewegen.
Das Spiel endet, wenn einer der 4 Walkers die rechte Bildschirmseite erreicht hat, und somit die Basis zerstören kann. Das Spiel ist ebenfalls beendet, wenn man alle Snowspeeder verloren hat. Gesteuert wird mit Joystick 1.

Bitte lesen Sie weiter auf Seite 23

```
10 ! ********************
11 ! *      STARWAR2      *
12 ! *    Das Imperium    *
13 ! *  schlaegt zurueck  *
14 ! *                    *
15 ! *    Copyright by    *
16 ! *                    *
17 ! *  Patrick  Schmitz  *
18 ! *                    *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole  *
21 ! *     Ext. Basic     *
22 ! *     Joystick 1     *
23 ! * (opt. Speechsynth.)*
24 ! *                    *
26 ! *  Speicherbelegung  *
27 ! *    10827 Bytes     *
28 ! *                    *
29 ! ********************

100 CALL PEEK(-28672,Z):: IF
 Z<>96 THEN SP=0 ELSE SP=-1
:: CALL SPGET("START",A$)
110 !
120 !Star Wars Melodie
130 !
140 DATA 175.14,262.15,233.0
6,220.06,196.06,349.14,262.1
5,233.06,220.06,196.06,349.1
4,262.15,233.06,220.06,233.0
6,196.20,0
150 !
160 !Titelbild
170 !
180 CALL CLEAR :: CALL SCREE
N(2):: CALL DELSPRITE(ALL)::
 CALL CHARSET :: FOR Z=1 TO
48 :: CALL HCHAR(INT(RND*24)
+1,INT(RND*32)+1,46):: NEXT
Z :: CALL COLOR(2,16,1):: DI
M ATAT(5):: ON BREAK NEXT ::
 RANDOMIZE
190 FOR Z=65 TO 90 :: CALL C
HARPAT(Z,B$):: B$="0000"&SEG
$(B$,1,4)&SEG$(B$,7,4)&SEG$(
B$,13,4):: CALL CHAR(Z+32,B$
):: NEXT Z
200 CALL CHAR(107,"000000242
8302824",115,"0000003C403804
78"):: IF SP THEN IF RND<.7
THEN CALL SAY("#WELL##COME")
:: CALL SAY("IN SPACE")ELSE
CALL SAY("HELLO AGAIN")
210 CALL COLOR(3,4,2,4,4,2):
: FOR Z=5 TO 8 :: CALL COLOR
(Z,8,2):: CALL COLOR(Z+4,4,2
):: NEXT Z :: CALL MAGNIFY(2
):: Q=.1
220 CALL CHAR(96,"3C4299A1A1
99423C"):: DISPLAY AT(1,1):"
THE STAR WARS SAGA CONTINUES
" :: DISPLAY AT(24,1):"` 198
4..patrick schmitz"
```

```
230 CALL TITEL("krieg",7,7,0
,12):: CALL TITEL("der",9,10
,5,12):: CALL TITEL("sterne"
,13,6,8,12):: CALL TITEL("TE
IL 2",19,7,18,9)
240 IF SP THEN CALL SAY(,SEG
$(A$,1,60),"###WANTS"):: CAL
L SAY("PART TWO")
250 CALL STARWARS(2600,6)::
DISPLAY AT(1,1):"" :: DISPLA
Y AT(24,1):"" :: CALL SCREEN
(12):: CALL DELSPRITE(ALL)::
 CALL SCREEN(7):: CALL SCREE
N(2)
260 CALL TITEL("DAS",3,10,0,
11):: CALL TITEL("IMPERIUM",
10,6,3,11):: CALL TITEL("SCH
LAEGT",17,6,11,11):: CALL TI
TEL("ZURUECK",23,8,19,11)
270 CALL STARWARS(2400,3)::
CALL MOTION(#1,-100,100):: F
OR Z=2 TO 28 :: CALL MOTION(
#Z,-100,100):: CALL DELSPRIT
E(#Z-1):: NEXT Z
280 CALL KEY(0,KEY,ST) :: IF
 KEY=13 THEN X=16 :: CALL SO
UND(-50,1400,3,1410,7) :: CA
LL SOUND(-50,1400,5,1410,9)E
LSE X=5
290 !
300 !Anleitung
310 !
320 CALL CLEAR :: CALL SCREE
N(2):: CALL COLOR(1,15,16)::
 FOR Z=2 TO 14 :: CALL COLOR
(Z,X,16):: NEXT Z :: IF X=16
 THEN 420
330 DISPLAY AT(1,1):"IN EINE
R GALAXIS, WEIT, WEITENTFERN
T, BRACH DER KRIEG...DER STE
RNE AUS. DEN REBELLENWAR ES
GELUNGEN, DEN TODES- STERN U
ND DAMIT DAS ZENTRUM"
340 DISPLAY AT(6,1):"DER BOE
SEN MAECHTE ZU ZER-..STOEREN
. DOCH DARTH VADER,..DER DAR
K LORD DER SITH, IST ENTKOMM
EN. ER VERFOLGT MIT..HILFE D
ER IMPERIALEN STURM-"
350 DISPLAY AT(11,1):"TRUPPE
N DIE MITGLIEDER DER..ALLIAN
Z DER REBELLEN. DIESE HABEN
SICH AUF DEN EISPLA-..NETEN
HOTH ZURUECKGEZOGEN."
360 DISPLAY AT(15,1):"NUN GR
EIFEN DIE 11 METER....GROSSE
N AT-ATS, DIE IMPERIALWALKER
S, DIE RIESIGEN MECHA-NISCHE
N MONSTER DES BOESEN..IMPERI
UMS AN."
370 DISPLAY AT(20,1):"LUKE,
DU UND DEINE FREUNDE..FLIEGE
N NUN DIE SNOWSPEEDER UND VE
```

LISTINGS

```
RSUCHEN DIE AT-ATS....AUFZUH
ALTEN, BEVOR SIE DIE..BASIS
ERREICHEN."
380 IF SP THEN IF RND<.5 THE
N CALL SAY("PLEASE WAIT JUST
 A1 SECOND")ELSE CALL SAY("P
LEASE READ THE INSTRUCTIONS"
)
390 !
400 !Zeichendef.
410 !
420 CALL DELSPRITE(ALL):: CA
LL MAGNIFY(4):: CALL CHAR(14
0,"0000071F1F3F3F0808080C040
40E0000000080EEFFEEE0808080C
04040E00000")
430 CALL CHAR(136,"000000000
0187018"&RPT$("0",48)):: CAL
L CHAR(132,RPT$("0",16)&"000
000001092E5FB"&RPT$("0",16)&
"000000000042537BFF")
440 CALL CHAR(128,RPT$("0",1
6)&"0000000091D5FBFF"&RPT$("
0",23)&"148A9ABFF"):: CALL C
HAR(124,RPT$("0",11)&"607171
71F5F5F7FFFFFFF"&RPT$("0",13
)&"8080C0C1C9E9E9F9FFF")
450 CALL CHAR(120,"000000000
101010303131737373F3F7FFF00008
08080C0C0E0E0E0F4F4FEFFFFFF"
):: CALL CHAR(104,RPT$("F",1
6)):: CALL CHAR(112,RPT$("F"
,16))
460 CALL CHAR(116,"000000000
00007"&RPT$("0",48)):: CALL
CHAR(108,"000000000144010"&
RPT$("0",48)):: CALL CHAR(10
5,"FFFFFFFFFFFEA6A4")
470 CALL CHAR(100,"004200140
0401200"&RPT$("0",48)):: CAL
L CHAR(96,"0280140040100100"
&RPT$("0",48)):: CALL CHAR(9
2,"8002000000002001"&RPT$("0
",48))
480 CALL CHAR(60,RPT$("0",16
)&"01070F1F3F39081D"&RPT$("0
",16)&"0CEEFCE0000080C0")::
CALL CHAR(36,"00000000000415
3D3F7F7FFFFFFFFFFF0000004060
E0E0F0F0F9F9FBFFFFFFFF")
490 CALL CHAR(40,"0000000000
0000080C0C0D1D9FBFFFFF000000
0008183838BCBDBDFFFFFFFFFF")
500 !
510 !Schwierigkeitsgrad
520 !
530 IF X=16 THEN 610 ELSE DI
SPLAY AT(24,23)SIZE(6):"WEIT
ER" :: CALL KEY(0,KEY,ST)::
IF ST=0 THEN DISPLAY AT(24,2
3)SIZE(6):"" :: GOTO 530
540 IF KEY=49 THEN Q=.1 ELSE
 IF KEY=50 THEN Q=.16 ELSE I
```

```
F KEY=51 THEN Q=.23 ELSE 570
550 CALL SOUND(-100,KEY*10,3
,KEY*10+2,9):: CALL SOUND(-1
00,KEY*11,6,KEY*11+2,14)
560 IF NOT SP THEN 570 ELSE
IF KEY=49 THEN CALL SAY("ONE
")ELSE IF KEY=50 THEN CALL S
AY("TWO")ELSE IF KEY=51 THEN
 CALL SAY("THREE")
570 CALL CLEAR :: CALL DELSP
RITE(ALL):: FOR Z=1 TO 14 ::
 CALL COLOR(Z,16,16):: NEXT
Z
580 !
590 !Screen-Aufbau
600 !
610 CALL CHAR(44,"00000001"&
RPT$("0",56)):: FOR Z=8 TO 2
0 :: CALL SPRITE(#Z,44,1,INT
(RND*100)+40,INT(RND*256)+1,
0,-2):: NEXT Z
620 FOR Z=1 TO 3 :: CALL HCH
AR(Z,1,104,32):: NEXT Z :: C
ALL HCHAR(4,1,105,32) ::FOR
Z=23 TO 24 :: CALL HCHAR(Z,1
,112,32):: NEXT Z :: CALL HC
HAR(22,1,135,32)
630 DISPLAY AT(1,1)SIZE(28):
"hhTHEhEMPIREhSTRIKEShBACKhh
h" :: DISPLAY AT(24,1)SIZE(2
8):"PUNKTEp000ppppppppSCHIF
FEp3"
640 !
650 !Berge-Sprites
660 !
670 A$="120124036040" :: FOR
 Z=1 TO 4 :: CALL SPRITE(#20
+Z,VAL(SEG$(A$,Z*3-2,3)),1,5
,(Z*8-5)*8-7,0,-2):: NEXT Z
680 FOR Z=1 TO 4 :: CALL SPR
ITE(#24+Z,128+((Z/2-INT(Z/2)
=0)*-4),1,145,(Z*8-5)*8-7,0,
-2):: NEXT Z :: FOR Z=2 TO 8
 :: CALL COLOR(Z,16,5):: NEX
T Z
690 !
700 !Screen sichtbar
710 !
720 CALL COLOR(10,8,16,12,8,
16,11,15,16,13,15,16):: FOR
Z=21 TO 24 :: CALL COLOR(#Z,
16,#Z+4,15):: NEXT Z :: FOR
Z=8 TO 20 :: CALL COLOR(#Z,1
5):: NEXT Z
730 !
740 !Spielvorbereitung
750 !
760 IF SP THEN CALL SAY("#RE
ADY TO START")
770 FOR Z=1 TO 4 :: CALL SPR
ITE(#Z+1,140,2,5+Z*32,17,0,1
) :: ATAT(Z+1)=0 :: NEXTZ ::
 CALL SPRITE(#1,136,5,89,233
):: P=12 :: TI,TR,PU=0 :: SC
H=3
780 !
790 !Spielbeginn
800 !
810 FOR Z=2 TO 5 :: IF ATAT(
Z)=100 THEN 820 ELSE CALL PO
SITION(#Z,Y,Y):: IF Y<201 TH
EN 820 ELSE 950
820 NEXT Z
830 IF (P-1)/4<>INT((P-1)/4)
OR ATAT((P-1)/4+1)=100 THEN
1370 ELSE AT@=(P-1)/4+1
:: CALL KEY(1,K,S):: IF K=18
 THEN 1100 ELSE IF RND>.5 TH
EN 1420 ELSE CALL POSITION(#
AT@,X,Y)
840 !
850 !Schuss AT-AT
860 !
870 CALL SPRITE(#7,116,11,P*
8-7,Y,0,80):: CALL SOUND(-10
0,330,0,332,4):: CALL SOUND(
-100,332,2,334,6):: CALL SOU
ND(-100,334,4,336,8)
880 CALL DELSPRITE(#7):: FOR
 Z=1 TO 5 :: CALL SOUND(-100
,110,30,110,30,1000+Z*12,30,
-8,2):: CALL COLOR(#1,11)::
CALL COLOR(#1,7):: NEXT Z
890 CALL PATTERN(#1,108):: F
OR Z=1 TO 3 :: CALL SOUND(-1
00,-7,Z+3):: CALL PATTERN(#1
,104-Z*4):: NEXT Z :: CALL C
OLOR(#1,1)
900 SCH=SCH-1 :: DISPLAY AT(
24,28)SIZE(1):STR$(SCH):: IF
 SCH=0 THEN 950 ELSE CALL SO
UND(-100,120,2,122,4):: CALL
 SOUND(-100,118,4,120,6)
910 CALL SOUND(-100,116,6,11
8,8):: CALL PATTERN(#1,136):
: CALL COLOR(#1,5):: GOTO 14
20
920 !
930 !Keine Schiffe mehr
940 !
950 FOR Z=2 TO 5 :: CALL MOT
ION(#Z,0,0):: NEXT Z :: FOR
Z=1 TO 3 :: FOR Z1=1 TO
10 :: CALL SOUND(-100,110,Z1
,112,Z1+5):: NEXT Z1 :: NEXT
 Z
960 IF SP THEN CALL SAY("OH
NO"):: IF RND>.5 THEN CALL S
AY("OH NO")ELSE CALL SAY("NO
T VERY GOOD")
970 CALL DELSPRITE(ALL):: CA
LL CLEAR :: CALL CHARSET ::
FOR Z=1 TO 14 :: CALL COLOR(
Z,12,2):: NEXT Z
980 CALL MAGNIFY(1):: FOR Z=
1 TO 20 :: CALL SPRITE(#Z,46
```

```
,16,INT(RND*190)+1,1,0,INT(R
ND*12)+1):: NEXT Z
990 IF PU>HI THEN HI=PU
1000 DISPLAY AT(1,1):"LUKE,
ES IST DIR LEIDER": :"NICHT
GELUNGEN, DIE AT-ATs": :"AUF
ZUHALTEN.": :"DOCH DER": :"K
RIEG DER  S T E R N E"
1010 DISPLAY AT(11,1):"GEHT
WEITER...":TAB(22);"4 >ENDE"
: :TAB(22);"8 >REDO": :PU;"P
UNKTE" :: FOR Z=200 TO 0 STE
P -1 :: CALL KEY(0,KEY,ST)::
 DISPLAY AT(13,15):Z
1020 IF KEY=56 OR KEY=6 THEN
 Z=0 :: GOTO 1050 ELSE IF KE
Y<>52 AND KEY<>2 THEN 1050 E
LSE CALL CLEAR :: CALL DELSP
RITE(ALL):: CALL CHARSET ::
PRINT "HIGHSCORE :";HI: : :
: :"DER KRIEG DER S T E R N
E"
1030 PRINT :"IST NICHT BEEND
ET.": :"DAS  I M P E R I U M
..WIRD": :"WIEDER ZURUECKSCH
LAGEN.": :"BIS ZUM NAECHSTEN
 MAL...": :"DIE M A C H T SE
I MIT DIR !!": : : :
1040 IF NOT SP THEN STOP ELS
E CALL SAY("BYE BYE"):: END
1050 NEXT Z :: CALL SOUND(-5
0,1400,0,1410,8):: CALL CLEA
R :: FOR Z=1 TO 14 :: CALL C
OLOR(Z,16,16):: NEXT Z :: CA
LL COLOR(1,15,16):: X=16 ::
CALL DELSPRITE(ALL)
1060 IF NOT SP THEN 420 ELSE
 IF RND>.5 THEN CALL SAY("#O
#K"):: CALL SAY("#TRY AGAIN"
):: GOTO 420 ELSE CALL SAY("
O K AN OTHER TRY"):: GOTO 42
0
1070 !
1080 !AT-AT getroffen
1090 !
1100 CALL POSITION(#AT@,X,Y)
:: CALL SPRITE(#7,116,7,P*8-
7,217,0,-(Y*-.5+115))
1110 FOR Z=1 TO 3 :: CALL SO
UND(-100,440+Z*2,Z*2,442+Z*2
,Z*3):: NEXT Z
1120 CALL DELSPRITE(#7):: FO
R Z=1 TO 3 :: CALL COLOR(#AT
@,7):: CALL SOUND(-50,-6,Z*2
):: CALL COLOR(#AT@,2):: NEX
T Z
1130 ATAT(AT@)=ATAT(AT@)+1 :
: IF ATAT(AT@)<10 THEN 1320
ELSE CALL MOTION(#AT@,0,,0)
1140 FOR Z=1 TO 6 :: CALL CO
LOR(#AT@,11):: CALL COLOR(#A
T@,7):: CALL SOUND(-100,110,
30,110,30,2000+Z*20,30,-8,1)
```

LISTINGS

```
:: NEXT Z
1150 CALL COLOR(#AT@,2):: CA
LL PATTERN(#AT@,108):: CALL
PATTERN(#AT@,60) :: CALL SOU
ND(-100,-6,2)::CALL COLOR(#A
T@,7):: CALL COLOR(#AT@,12):
: CALL SOUND(-100,-6,4)
1160 CALL PATTERN(#AT@,100):
: CALL COLOR(#AT@,2):: CALL
PATTERN(#AT@,96) :: CALL SOU
ND(-200,-5,3)::CALL DELSPRIT
E(#AT@):: PU=PU+1000 :: CALL
 PUNKTE(PU):: ATAT(AT@)=100
1170 IF NOT SP THEN 1180 ELS
E IF RND<.2 THEN CALL SAY("#
NICE TRY")ELSE IF RND>.8 THE
N CALL SAY("#GOOD WORK")
1180 TR=TR+1 :: IF TR<4 THEN
 1420 ELSE DISPLAY AT(12,1):
".DIE.M.A.C.H.T.IST.MIT.DIR.
" :: CALL STARWARS(2700,3)::
 CALL CLEAR :: CALL DELSPRIT
E(ALL):: CALL CHARSET :: PU=
PU+SCH*1000+(200-TI)*20
1190 IF PU>HI THEN HI=PU
1200 !
1210 !Alle AT-ATs getroffen
1220 !
1230 CALL MAGNIFY(1):: FOR Z
=1 TO 20 :: CALL SPRITE(#Z,4
6,16,INT(RND*190)+1,1,0,INT(
RND*12)+1):: NEXT Z :: FOR Z
=1 TO 14 :: CALL COLOR(Z,12,
2):: NEXT Z
1240 DISPLAY AT(1,1):"GRATUL
IERE, LUKE, DU HAST": :"DIE
AT-ATs AUFGEHALTEN.": :"HAN
SOLO, LEIA ORGANA,": :"CHEWI
, R2D2 UND C-3PO FREUEN": :"
SICH. OBI WAN INSTRUIERT"
1250 DISPLAY AT(11,1):"DICH:
 DU MUSST INS DAGOBAH-": :"S
YSTEM ZU JODA, DEM MEISTER":
 :"ALLER JEDI-RITTER FLIEGEN
,": :"DER DICH ZUM JEDI MACH
EN": :"WIRD..."
1260 DISPLAY AT(22,22):"4 >E
NDE" :: DISPLAY AT(24,1):PU;
"PUNKTE";TAB(22);"8 >REDO" :
: FOR Z=250 TO 0 STEP -1 ::
DISPLAY AT(23,15):Z :: CALL
KEY(0,KEY,ST)
1270 IF KEY=56 OR KEY=6 THEN
 Z=0 :: GOTO 1300 ELSE IF KE
Y<>52 AND KEY<>2 THEN 1300 E
LSE CALL CLEAR :: CALL CHARS
ET :: CALL DELSPRITE(ALL)
1280 PRINT "HIGHSCORE :";HI:
 : : : :"DIE  J E D I  RITTE
R WERDEN": :"ZURUECKKEHREN !
!!": :"BIS ZUM NAECHSTEN MAL
...": :"DIE M A C H T SEI MI
T DIR !!": : : :
```

```
1290 IF NOT SP THEN STOP ELS
E CALL SAY("SEE YOU IN SPACE
"):: CALL SAY("GOODBYE"):: E
ND
1300 NEXT Z :: CALL SOUND(-5
0,1400,0,1410,2):: CALL CLEA
R :: CALL DELSPRITE(ALL):: C
ALL COLOR(1,15,16):: FOR Z=2
 TO 14 :: CALL COLOR(Z,16,16
):: NEXT Z :: X=16
1310 IF NOT SP THEN 420 ELSE
 IF RND<.5 THEN CALL SAY("O
K WHY NOT"):: GOTO 420 ELSE
CALL SAY("BE SO GOOD AS LAST
 TIME"):: GOTO 420
1320 IF ATAT(AT@)/2=INT(ATAT
(AT@)/2)THEN CALL MOTION(#AT
@,0,2)ELSE CALL MOTION(#AT@,
0,0)
1330 PU=PU+100 :: CALL PUNKT
E(PU):: GOTO 1420
1340 !
1350 !Schuss
1360 !
1370 CALL KEY(1,K,K):: IF K<
>18 THEN 1420 ELSE CALL SPRI
TE(#7,116,7,P*8-7,217,0,-120
):: CALL SOUND(-100,440,0,44
2,4)
1380 CALL SOUND(-100,442,2,4
44,6):: CALL SOUND(-100,444,
4,446,8):: CALL SOUND(-100,4
44,4,446,8):: CALL DELSPRITE
(#7)
1390 !
1400 !Bewegung AT-AT
1410 !
1420 IF RND>Q THEN 1460 ELSE
 X=INT(RND*4)+2 :: IF ATAT(X
)=100 THEN 1460 ELSE CALL MO
TION(#X,0,1)
1430 !
1440 !Haupt-Joyst-Abfrage
1450 !
1460 CALL JOYST(1,X,Y):: Y=-
SGN(Y):: IF P+Y<5 OR P+Y>18
THEN 810 ELSE P=P+Y :: CALL
LOCATE(#1,P*8-7,233):: TI=TI
+1 :: ON (TI<30)+2 GOTO 830,
810
1470 !
1480 !
1490 !Unterprogramme
1500 !
1510 SUB PUNKTE(PU):: IF PU<
0 THEN PU=0
1520 DISPLAY AT(24,8)SIZE(LE
N(STR$(PU))):STR$(PU):: SUBE
ND
1530 !
1540 !Anzeige TitelSprites
1550 !
1560 SUB TITEL(A$,X,Y,Z1,Z2)
:: FOR Z=1 TO LEN(A$):: CALL
 SPRITE(#Z1+Z,ASC(SEG$(A$,Z,
1)),Z2,(X*8-7)-Z*5.4,(Y*8-7)
+Z*20):: NEXT Z :: SUBEND
1570 !
1580 !Star Wars Melodie
1590 !
1600 SUB STARWARS(X,Y):: RES
TORE 140 :: FOR Z=1 TO 16
1610 READ Z1 :: CALL SOUND(5
00,Z1*2,2,Z1*2+Y,4,Z1,6):: C
ALL SOUND((Z1-INT(Z1))*-X,Z1
*2,4,Z1*2+Y,8,Z1,10):: NEXT
Z :: SUBEND
```

Fortsetzung von Seite 19

Variablen-Liste

Z,Z1	allgemein verwendet
X,Y	allgemein/in CALL JOYST
Key,K,ST	in CALL KEY verwendet
SCH	Schiffe
PU	Punkte
P	Zeilenposition von #1 (Snowspeeder)
ATAT(2–5)	#2–#5(AT-AT) Treffer (100=zerstört)
AT	=AT-AT(P-1)/4+1 (bestimmtes AT-AT)
TR	Anzahl zerstörter AT-ATs
SP	Sprachsynth? ja= -1 nein=0
HI	Highscore
Q	Schwierigkeitsgrad (Standard=.1)

Sprites:

# 1	Snowspeeder
# 2–5	AT-ATs
# 7	Laser-Schuß
# 8–20	Schnee
# 21–24	Berge
# 25–28	Boden

Bedienung:

Sobald unter der Anleitung “WEITER“ blinkt, “1“ für einfach, “2“ für mittelschwer oder “3“ für schwer drücken. Wird eine andere Taste gedrückt, nimmt der Computer automatisch “einfach“ an.
Wenn man die Spielanleitung übergehen will, kann man, sobald die Star Wars-Melodie zum zweitenmal erklungen ist, die ENTER-Taste gedrückt halten, bis sich der Bildschirm weiß färbt. Der Schwierigkeitsgrad ist dann allerdings immer “einfach“.

Programmaufbau

10– 29	REMs (c)
100– 110	Sprachsynthesizer???
120– 150	DATAs für Krieg der Sterne-Melodie
160– 250	Titelbild 1 (Star Wars – Krieg der Sterne II)
260– 290	Titelbild 2 (Empire – Imperium...)
300– 390	Anleitung, Beschreibg. Star Wars II
400– 500	Zeichendefinitionen (Walkers, ...)
510– 580	Wahlweise Schwierigkeitsgrad
590– 690	Aufbau des Screens
700– 730	Farbzuordnungen
740– 780	Snowspeeder/Walker/Variablen=0 setzen
790– 840	AT-AT (=Walker) rechts???
850– 920	AT-AT schießt auf Snowspeeder
930– 990	AT-AT rechts oder kein Schiff mehr
1000–1040	Spielende
1050–1070	neues Spiel
1080–1130	AT-AT getroffen
1140–1200	AT-AT zerstört (zehn Treffer)
1210–1270	alle AT-ATs zerstört
1280–1290	Spielende
1300–1310	neues Spiel
1320–1340	AT-AT bewegen (zufällig)

YATHZEE

Dieses Spiel für 1–6 Personen ist auch unter dem Namen Kniffel bekannt. Bei jedem Durchgang kann maximal dreimal gewürfelt werden, wobei der erste Wurf mit allen fünf Würfeln durchgeführt wird (Feuertaste von Joystick 1). Nach jedem Wurf kann entschieden werden, welche Würfel nochmal benutzt werden, indem man, wenn der schwarze Pfeil auf einen Würfel zeigt, die Feuertaste betätigt. Will man einen Würfel gelten lassen, führt man den Pfeil durch den Joystickknüppel nach rechts weiter. Wenn der Pfeil den rechten Bildschirmrand erreicht, kommt der nächste Wurf mit den ausgewählten Würfeln. Falls kein Würfel aussortiert wurde, erfolgt die Aufforderung, den Wurf in die Gewinnkarte einzutragen. Selbst wenn kein Gewinnwurf erreicht wurde, muß ein Eintrag erfolgen, in diesem Falle eben Null Punkte. Der Eintrag erfolgt, indem man zuerst die Zeile, dann die Spalte, festlegt (Steuerung der Pfeile mit dem Joystick). Die Gewinnkarte ist in zwei Bereiche aufgeteilt. Im oberen Teil werden Einer, Zweier, . . . , Sechser eingetragen. Bei einem Wurf zählen nur die Würfel mit gleicher Punktzahl und deren Summe wird in das entsprechende Kästchen eingetragen. Erzielt ein Spieler im oberen Teil mehr als 62 Punkte, so erhält er einen Bonus von 35 Punkten gutgeschrieben.
Im unteren Teil werden Kombinationen eingetragen. Dreier- und Viererpasch bedeutet drei, bzw. vier gleiche Würfel. Die Punkte aller Würfel werden im Kästchen eingetragen! Full House sind drei gleiche Zahlen einer Art und zwei gleiche Zahlen einer anderen Art. Kleine Straße ist eine Folge von vier Würfeln, der fünfte eine beliebige Augenzahl. Große Straße ist eine Folge von fünf Würfeln aufeinanderfolgender Zahlen. Yahtzee heißt ein Wurf aus lauter gleichen Zahlen. Für diese Kombinationen werden feste Punktzahlen gutgeschrieben (s. Gewinnkarte). Chance eröffnet dem Spieler die Möglichkeit, einen Wurf aufzuschreiben, der keine anderen Bedingungen erfüllt oder dessen Kästchen alle schon belegt sind.
Wirft ein Spieler Yahtzee öfter als er in die Felder eintragen kann, und sind die entsprechenden Felder im oberen Teil der Gewinnkarte ebenfalls belegt, so kann er ihn im unteren Teil in ein beliebiges Feld eintragen (Joker).

Das Spiel gewinnt nicht unbedingt derjenige, der als erster alle Kästchen gefüllt hat, sondern wer die höchste Gesamtpunktzahl erreicht, also am geschicktesten kombiniert.

DER HEXER

Mit dem Hexer hat man ein sehr wirkungsvolles Mittel zur Zahlenumwandlung in der Hand (oder besser: im Computer).
Es ist leicht zu bedienen und bietet die Möglichkeit zur Umrechnung von fast allen wichtigen Zahlensystemen (sogar die etwas in Vergessenheit geratenen BCD-Zahlen wurden berücksichtigt).
Nach Start des Programms erscheint zuerst das Titelbild und schließlich das Wahlmenü. Hier kann man zwischen 6 Funktionen wählen, welche nach dem gleichen Schema ablaufen:
- (kurzes Warten)
- Ausgabe der umgerechneten Zahl
- Taste drücken

Als wichtigstes wäre noch zu erwähnen, wie groß die einzugebenden Zahlen maximal sein dürfen: Bei Dezimalzahlen maximal 268435455 (=2^28=28 Bits), bei Binär- und Hexadezimalzahlen genau eine Bildschirmzeile.



Überschreitungen werden nicht angenommen, aber auch Falscheingaben werden bemerkt. Gibt man eine negative Dezimalzahl ein, dann macht einen der Computer darauf aufmerksam und addiert den Wert 65536, um das Komplement zu erhalten.
Noch eine Bemerkung zu den BCD-Zahlen, die wohl nur wenige der Leser kennen werden: Zerlegt man eine Dezimal- oder Hexdezimalzahl in einzelne Ziffern, übersetzt dann diese Ziffern in den Binärcode und fügt den Code wieder in Reihenfolge der Ziffern zusammen, so erhält man die entsprechende BCD-Zahl (nimmt man als Ausgangsbasis eine Hexzahl, so erhält man natürlich als Endprodukt die dazugehörige Binärzahl, die dem BCD-Code entspricht). Deshalb müssen eingegebene BCD-Zahlen in ihrer Länge immer durch 4 teilbar sein, da jeder Dezimalziffer 4 Binärziffern entspricht.

LISTINGS

```
10 ! *********************
11 ! *      YAHTZEE       *
12 ! *    (Vers.1.4.)     *
13 ! *                    *
14 ! *    Copyright by    *
15 ! *                    *
16 ! *    Roger Hueser    *
17 ! *                    *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole  *
21 ! *     Ext. Basic     *
22 ! *     Joystick 1     *
23 ! *    Cassettenrec.   *
24 ! *(oder Disk+32K-Erw.)*
25 ! *                    *
26 ! *  Speicherbelegung  *
27 ! *     9212 Bytes     *
28 ! *                    *
29 ! *********************
100 GOTO 110 :: CALL PAUSE :
: CALL CHAR :: CALL CHARPAT
:: CALL CHARSET :: CALL CLEA
R :: CALL COLOR :: CALL HCHA
R :: CALL INIT :: CALL JOYST
 :: CALL KEY :: CALL SOUND :
: CALL VCHAR
110 ON BREAK NEXT :: CALL CL
EAR :: ON WARNING NEXT :: OP
TION BASE 1 :: CALL SCREEN(4
):: RANDOMIZE
120 DIM HI(5),HI$(5),SPIELER
$(6),SPERR(6,3),KEN(5),ERGEB
$(6,16,3),ZW(6,3),ER(6,3),W(
6)
130 GOTO 150 :: A$,B$,C$,D$,
Q$
140 SPIEL,SPIELER,SUM,HILS,D
G,SX,SY,YS,XS,I,C,K,S,R,M,H,
P,U,Y,X,Q
150 !@P-
160 GOTO 230
170 CALL HCHAR(1,H*3+14,122)
:: CALL HCHAR(2,H*3+14,129):
: CALL HCHAR(1,H*3+15,130)::
 CALL HCHAR(2,H*3+15,123)::
GOTO 490
180 CALL HCHAR(1,H*3+14,124)
:: CALL HCHAR(2,H*3+14,125):
: CALL HCHAR(1,H*3+15,126)::
 CALL HCHAR(2,H*3+15,127)::
GOTO 490
190 CALL HCHAR(1,H*3+14,128)
:: CALL HCHAR(2,H*3+14,129):
: CALL HCHAR(1,H*3+15,130)::
 CALL HCHAR(2,H*3+15,131)::
GOTO 490
200 CALL HCHAR(1,H*3+14,132)
:: CALL HCHAR(2,H*3+14,133):
: CALL HCHAR(1,H*3+15,134)::
 CALL HCHAR(2,H*3+15,135)::
GOTO 490
210 CALL HCHAR(1,H*3+14,136)
:: CALL HCHAR(2,H*3+14,137):
: CALL HCHAR(1,H*3+15,138)::
 CALL HCHAR(2,H*3+15,139)::
GOTO 490
220 CALL HCHAR(1,H*3+14,140)
:: CALL HCHAR(2,H*3+14,141):
: CALL HCHAR(1,H*3+15,142)::
 CALL HCHAR(2,H*3+15,143)::
GOTO 490
230 CALL CHAR(122,RPT$("0",1
5)&"18"):: CALL CHAR(124,"00
00303"&RPT$("0",50)&"C0C")
240 CALL CHAR(128,"000030300
000000101"&RPT$("0",28)&"808
00000000C0C"):: CALL CHAR(13
2,"0000303000000000000000003
0300000000000C0C000000000000
0000C0C")
250 CALL CHAR(136,"000030300
0000001010000003030000000000
C0C0000000808000000000C0C")::
CALL CHAR(140,"0000303000000
0303000000000303000000000C0C0
000000C0C000000000C0C")
260 FOR H=65 TO 87 :: CALL C
HARPAT(H,A$):: CALL CHAR(H+3
2,A$):: NEXT H :: CALL CHARP
AT(32,A$,33,B$,63,C$,90,D$):
: CALL CHAR(111,A$):: CALL C
HAR(118,D$):: CALL CHAR(106,
C$)
270 CALL CHAR(98,B$):: CALL
CHAR(120,"F7FBFD0000FDFBF7E7
E7E7E766A5C3E7"):: DISPLAY A
T(13,1)BEEP ERASE ALL:"WIEVI
ELE MITSPIELER ?"
280 ACCEPT AT(13,24)VALIDATE
("123456")SIZE(1):SPIELER
290 FOR H=1 TO SPIELER :: DI
SPLAY AT(10,1)BEEP ERASE ALL
:"NAME DES ";STR$(H);". MITS
PIELERS" :: ACCEPT AT(12,1)V
ALIDATE(UALPHA)SIZE(11):SPIE
LER$(H):: NEXT H :: CALL CLE
AR
300 CALL CHAR(64,"000000F392
F282F2"):: CALL CHAR(94,"008
8888888888EE"):: CALL CHAR(
92,"0040E0464444444465"):: CAL
L CHAR(36,"000000FFFF")
310 CALL CHAR(39,"E070381818
18181818181818183870E0070E1C
1818181818181818181C0E07")
:: CALL CHAR(58,"1818181F1F0
000000000000FFFF181818181818F
8F8")
320 CALL CHAR(37,"1818181818
181818"):: CALL CHAR(38,"181
818FFFF"):: CALL CHAR(35,"00
00001"):: CALL CHAR(33,"0008
0C7E7E0C08"):: CALL CHAR(34,
"00020202F29292FB")
330 FOR S=1 TO SPIELER :: FO
R R=1 TO 13 :: FOR M=1 TO 3
```

```
:: ERGEB$(S,R,M)="--" :: NEX
T M :: NEXT R :: NEXT S :: F
OR S=1 TO SPIELER :: FOR R=1
4 TO 15 :: FOR M=1 TO 3 :: E
RGEB$(S,R,M)=" 0" :: NEXT M
:: NEXT R :: NEXT S :: FOR S
=1 TO SPIELER
340 FOR M=1 TO 3 :: ERGEB$(S
,16,M)="..0" :: ZW(S,M),ER(S
,M),SPERR(S,M)=0 :: NEXT M :
: NEXT S :: FOR H=12 TO 14 :
: CALL COLOR(H,16,2):: CALL
COLOR(H-3,15,9):: NEXT H
350 DISPLAY AT(5,16)ERASE AL
L:"M""1..M""2..M""3" :: CALL
 HCHAR(3,1,36,32):: CALL VCH
AR(1,15,37,2):: CALL HCHAR(3
,15,38)
360 DISPLAY AT(7,1):" EINSER
########":" ZWEIER########":
" DREIER########":" VIERER##
######":" FUENFER#######":"
SECHSER#######":" )GES.>62
 GES":" *BONUS 35("
370 DISPLAY AT(16,2):"3@PASC
H##AUGEN":" 4@PASCH##AUGEN":
" FU^ HOUSE###25":" KLEINE S
\###30":" GROSSE S\###40":"
YAHTZEE#####50":" CHANCE###A
UGEN": :" ENDSUMME"
380 FOR SPIEL=1 TO 39 :: FOR
 S=1 TO SPIELER :: FOR H=1 T
O 5 :: KEN(H)=0 :: NEXT H ::
 IF SPIELER=1 AND SPIEL>1 TH
EN 450 ELSE DISPLAY AT(1,1)S
IZE(12):SPIELER$(S)
390 FOR P=18 TO 30 :: CALL V
CHAR(7,P,32,18):: NEXT P
400 FOR P=1 TO 6 :: DISPLAY
AT(P+6,16):"#"&ERGEB$(S,P,1)
&"###"&ERGEB$(S,P,2)&"###"&E
RGEB$(S,P,3):: NEXT P
410 DISPLAY AT(13,16):"="&ER
GEB$(S,14,1)&"###"&ERGEB$(S,
14,2)&"###"&ERGEB$(S,14,3)
420 DISPLAY AT(14,17):ERGEB$
(S,15,1)&"###"&ERGEB$(S,15,2
)&"###"&ERGEB$(S,15,3)
430 FOR H=7 TO 13 :: DISPLAY
 AT(H+9,16):"#"&ERGEB$(S,H,1
)&"###"&ERGEB$(S,H,2)&"###"&
ERGEB$(S,H,3):: NEXT H
440 DISPLAY AT(24,16):ERGEB$
(S,16,1)&".."&ERGEB$(S,16,2)
&".."&ERGEB$(S,16,3)
450 CALL HCHAR(3,2,59):: CAL
L VCHAR(4,2,37,2):: CALL HCH
AR(6,2,58):: CALL HCHAR(6,3,
36,14):: CALL HCHAR(3,17,59)
:: CALL VCHAR(4,17,37,2)
460 CALL HCHAR(6,17,60):: DG
=0 :: DISPLAY AT(4,1)SIZE(14
):"wuerfelnjooooo" :: DISPLA
Y AT(5,1)BEEP SIZE(14):"drue
ckeotasteb"
470 CALL KEY(1,H,U):: IF H<>
18 THEN 470 ELSE CALL PAUSE
:: CALL HCHAR(3,2,36):: CALL
 HCHAR(3,17,36):: CALL HCHAR
(4,2,32,16):: CALL HCHAR(5,2
,32,16)
480 CALL HCHAR(6,2,32,16)::
FOR H=1 TO 5 :: IF KEN(H)=0
THEN W(H)=INT(RND*6)+1 :: ON
 W(H)GOTO 170,180,190,200,21
0,220
490 NEXT H
500 DG=DG+1 :: IF DG>2 THEN
DG=0 :: GOTO 590 ELSE X=16
510 H=3 :: CALL HCHAR(1,X,33
)
520 FOR P=1 TO 2+H :: CALL K
EY(1,Q,U):: IF Q=18 THEN KEN
((X-13)/3)=0 :: CALL HCHAR(1
,X,32,3):: CALL HCHAR(2,X,32
,3):: X=X+3 ELSE 540
530 CALL KEY(1,Q,U):: IF Q=1
8 THEN 530 ELSE 570
540 CALL JOYST(1,Q,U):: IF Q
=4 THEN CALL HCHAR(1,X,32)::
 :: KEN((X-13)/3)=1 :: X=X+3
 ELSE 560
550 CALL JOYST(1,Q,U):: IF Q
=4 THEN 550 ELSE 570
560 NEXT P :: IF H=3 THEN H=
0 :: CALL HCHAR(1,X,32):: GO
TO 520 ELSE 510
570 IF X<29 THEN 510
580 IF 0=KEN(1)OR 0=KEN(2)OR
 0=KEN(3)OR 0=KEN(4)OR 0=KEN
(5)THEN 480 ELSE DG=0
590 CALL HCHAR(3,2,59):: CAL
L VCHAR(4,2,37):: CALL HCHAR
(5,2,58):: CALL HCHAR(5,3,36
,13):: CALL HCHAR(3,16,59)::
 CALL VCHAR(4,16,37)
600 CALL HCHAR(5,16,60):: DI
SPLAY AT(4,1)SIZE(13)BEEP:"s
oeotovoeonob"
610 Y=16
620 H=6 :: CALL HCHAR(Y,3,12
0)
630 FOR P=1 TO 3+H :: CALL J
OYST(1,Q,U):: IF U=0 THEN 69
0 ELSE CALL HCHAR(Y,3,32)::
Y=Y-U/4
640 IF Y=13 THEN Y=16
650 IF Y=15 THEN Y=12
660 IF Y>22 THEN Y=7
670 IF Y<7 THEN Y=22
680 CALL JOYST(1,Q,U):: IF U
=0 THEN 620 ELSE 680
690 CALL KEY(1,Q,U):: IF Q<>
18 THEN 710 ELSE CALL HCHAR(
Y,3,120):: CALL PAUSE
700 IF Y<13 THEN R=Y-6 :: GO
```

LISTINGS

```
TO 720 ELSE R=Y-9 :: GOTO 72
0
710 NEXT P :: IF H>0 THEN H=
0 :: CALL HCHAR(Y,3,32):: GO
TO 630 ELSE 620
720 X=20
730 H=6 :: CALL HCHAR(4,X,12
1)
740 FOR P=1 TO 3+H :: CALL K
EY(0,Q,U):: IF Q=13 THEN CAL
L HCHAR(4,X,32):: CALL HCHAR
(Y,3,32):: GOTO 610
750 CALL JOYST(1,Q,U):: IF Q
=0 THEN 780 ELSE CALL HCHAR(
4,X,32):: X=X+Q*5/4 :: CALL
PAUSE
760 IF X>30 THEN X=20 ELSE I
F X<20 THEN X=30
770 GOTO 730
780 CALL KEY(1,Q,U):: IF Q<>
18 THEN 790 ELSE M=(X-15)/5
:: CALL HCHAR(4,X,121):: CAL
L PAUSE :: GOTO 800
790 NEXT P :: IF H>0 THEN H=
0 :: CALL HCHAR(4,X,32):: GO
TO 740 ELSE 730
800 IF ERGEB$(S,R,M)<>"--" T
HEN CALL HCHAR(4,X,32):: CAL
L HCHAR(Y,3,32):: GOTO 610
810 CALL HCHAR(3,2,36):: CAL
L HCHAR(3,16,36):: CALL HCHA
R(4,2,32,15):: CALL HCHAR(5,
2,32,15)
820 SUM=0 :: IF R>6 AND R<13
 THEN 860 ELSE IF R=13 THEN
850
830 FOR H=1 TO 5 :: IF W(H)=
R THEN SUM=SUM+R
840 NEXT H :: GOTO 880
850 SUM=W(1)+W(2)+W(3)+W(4)+
W(5):: GOTO 880
860 FOR SX=1 TO 4 :: FOR SY=
SX+1 TO 5 :: IF W(SX)>W(SY)T
HEN C=W(SX):: W(SX)=W(SY)::
W(SY)=C
870 NEXT SY :: NEXT SX :: ON
 R-6 GOTO 1150,1170,1190,121
0,1240,1260,32767
880 IF SUM=0 THEN SX=131 ::
U=2^(1/12)ELSE 900
890 FOR P=11 TO 1 STEP -2 ::
 SY=SX*U^P :: FOR Q=23-P TO
1 STEP -3 :: CALL SOUND(-100
,SY,Q):: NEXT Q :: NEXT P
900 IF SUM=50 THEN SX=131 ::
 U=2^(1/12)ELSE 920
910 FOR P=1 TO 10 STEP 2 ::
SY=SX*U^P :: FOR Q=1 TO 20 S
TEP 3 :: CALL SOUND(-100,SY,
Q):: NEXT Q :: NEXT P
920 HILS=1 :: IF SUM>9 THEN
ERGEB$(S,R,M)=STR$(SUM)ELSE
ERGEB$(S,R,M)=" "&STR$(SUM)
```

LISTINGS

```
930 IF R<7 THEN DISPLAY AT(R
+6,M*5+12)SIZE(2):ERGEB$(S,R
,M)ELSE DISPLAY AT(R+9,M*5+1
2)SIZE(2):ERGEB$(S,R,M):: GO
TO 970
940 ZW(S,M)=ZW(S,M)+SUM :: I
F ZW(S,M)>9 THEN ERGEB$(S,14
,M)=STR$(ZW(S,M))ELSE ERGEB$
(S,14,M)=" "&STR$(ZW(S,M))
950 DISPLAY AT(13,M*5+12)SIZ
E(2):ERGEB$(S,14,M)
960 IF SPERR(S,M)=0 AND ZW(S
,M)>62 THEN SPERR(S,M)=1 ::
ERGEB$(S,15,M)="35" :: ER(S,
M)=ER(S,M)+35 :: R=15 :: DIS
PLAY AT(14,M*5+12)SIZE(2):ER
GEB$(S,15,M)
970 ER(S,M)=ER(S,M)+SUM :: E
RGEB$(S,16,M)=STR$(ER(S,M)):
: IF ER(S,M)<100 THEN ERGEB$
(S,16,M)=" "&STR$(ER(S,M))
980 IF ER(S,M)<10 THEN ERGEB
$(S,16,M)="  "&STR$(ER(S,M))
990 DISPLAY AT(24,M*5+11)BEE
P SIZE(3):ERGEB$(S,16,M)
1000 CALL HCHAR(3,2,59):: CA
LL VCHAR(4,2,37,2):: CALL HC
HAR(6,2,58):: CALL HCHAR(6,3
,36,14):: CALL HCHAR(3,17,59
):: CALL VCHAR(4,17,37,2)
1010 CALL HCHAR(6,17,60):: D
ISPLAY AT(4,1)SIZE(14):"weit
erjoooooooo" :: DISPLAY AT(5,
1)SIZE(14):"drueckeotasteb"
1020 CALL KEY(1,Q,U):: IF Q<
>18 THEN 1020 ELSE CALL PAUS
E :: CALL HCHAR(3,2,36):: CA
LL HCHAR(3,17,36):: CALL HCH
AR(4,2,32,16)
1030 CALL HCHAR(5,2,32,16)::
 CALL HCHAR(6,2,32,16):: DIS
PLAY AT(1,14):: DISPLAY AT(2
,14):: CALL HCHAR(4,X,32)::
CALL HCHAR(Y,3,32):: CALL PA
USE :: NEXT S :: NEXT SPIEL
1040 CALL CLEAR :: CALL CHAR
SET
1050 FOR S=1 TO SPIELER :: W
(S)=ER(S,1)+2*ER(S,2)+3*ER(S
,3):: NEXT S
1060 FOR S=1 TO SPIELER-1 ::
 FOR P=S+1 TO SPIELER :: IF
W(S)<W(P)THEN Q=W(S):: W(S)=
W(P):: W(P)=Q :: Q$=SPIELER$
(S):: SPIELER$(S)=SPIELER$(P
):: SPIELER$(P)=Q$
1070 NEXT P :: NEXT S
1080 FOR S=1 TO SPIELER :: D
ISPLAY AT(2*S,1):SPIELER$(S)
:: DISPLAY AT(2*S,13):W(S);"
PUNKTE" :: NEXT S :: XS=131
:: K=2^(1/12):: FOR I=1 TO 1
0 STEP 2 :: YS=XS*K^I
```

```
1090 FOR P=1 TO 5 :: NEXT P
:: FOR Q=1 TO 20 STEP 4 :: C
ALL SOUND(-100,YS,Q):: NEXT
Q :: NEXT I :: FOR P=1 TO 90
0 :: NEXT P :: DISPLAY AT(23
,1)BEEP:"WEITER ?":"DANN DRU
ECKE TASTE !"
1100 FOR P=1 TO 2000 :: CALL
 KEY(1,K,S):: IF K=18 THEN 1
120
1110 NEXT P
1120 CALL PAUSE :: DISPLAY A
T(23,1):"NOCH EIN SPIEL ?":"
DANN DRUECKE TASTE !" :: FOR
 P=1 TO 2000 :: CALL KEY(1,Q
,U)
1130 IF Q=18 THEN CALL CLEAR
 :: GOTO 230 ELSE CALL KEY(0
,Q,U):: IF Q=78 OR Q=110 THE
N P=2000 ELSE IF U<>0 THEN C
ALL CLEAR :: GOTO 230
1140 NEXT P :: CALL CLEAR ::
 END
1150 IF W(1)=W(3)OR W(2)=W(4
)OR W(3)=W(5)THEN SUM=W(1)+W
(2)+W(3)+W(4)+W(5)
1160 GOTO 880
1170 IF W(1)=W(4)OR W(2)=W(5
)THEN SUM=W(1)+W(2)+W(3)+W(4
)+W(5)
1180 GOTO 880
1190 IF W(1)=W(2)AND W(4)=W(
5)AND(W(3)=W(4)OR W(3)=W(2))
THEN SUM=25
1200 GOTO 880
1210 IF W(1)=W(5)THEN SUM=30
 :: GOTO 880 ELSE IF W(2)=W(
3)OR W(3)=W(4)THEN C=W(3)::
W(3)=W(4):: W(4)=W(5):: W(5)
=C
1220 IF W(1)+1=W(2)AND W(2)+
1=W(3)AND W(3)+1=W(4)OR W(2)
+1=W(3)AND W(3)+1=W(4)AND W(
4)+1=W(5)THEN SUM=30
1230 GOTO 880
1240 IF W(1)+1=W(2)AND W(2)+
1=W(3)AND W(3)+1=W(4)AND W(4
)+1=W(5)OR W(1)=W(5)THEN SUM
=40
1250 GOTO 880
1260 IF W(1)=W(5)THEN SUM=50
1270 GOTO 880
1280 !@P+
1290 SUB PAUSE
1300 A=RND :: CALL KEY(1,K,S
):: IF S<>0 THEN 1300
1310 A=RND :: CALL JOYST(1,X
,Y):: IF X<>0 OR Y<>0 THEN 1
300
1320 SUBEND
```

LISTINGS

```
10 REM*********************
11 REM*    DER HEX-er      *
12 REM*                    *
13 REM*   Copyright by     *
14 REM*                    *
15 REM*  Martin Bannert    *
16 REM*                    *
17 REM*Benoetigte Geraete*
18 REM* TI99/4A  Konsole *
19 REM*                    *
23 REM* Speicherbelegung *
24 REM*     3559 Bytes     *
25 REM*********************
26 REM
100 REM *******BCD-ZAHLEN MU
ESSEN IN VIERERBLOECKEN EING
EGEBEN WERDEN ********
110 CALL CLEAR
120 DIM F$(15)
130 RESTORE
140 FOR I=0 TO 15
150 READ F$(I)
160 NEXT I
170 G$="0123456789ABCDEF"
180 REM
190 REM *TITELBILD*
200 REM
210 CALL CHAR(64,"FFFFFFFFFF
FFFFFF")
220 PRINT "@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@............
.......@@@@@@@@....DER H E X
 -er...@@@@@@@@";TAB(25);"@@
@@"
230 PRINT "@@@@@@@@@@@@@@@@@
@@@@@@@@@@@": : : : : : :
:
240 PRINT "...C BY BAXE PROG
RAMS":
250 FOR I=1 TO 700
260 NEXT I
270 CALL CLEAR
280 FOR I=2 TO 8
290 CALL COLOR(I,4,4)
300 NEXT I
310 REM
320 REM *MENUE*
330 REM
340 PRINT : :"..MENUE:":"..[
1]=DEZIMAL-BINAER":"..[2]=BI
NAER-DEZIMAL":"..[3]=HEXADEZ
IMAL-BCD":"..[4]=BCD-HEXADEZ
IMAL":"..[5]=HEX-DEZ":
350 PRINT "..[6]=DEZ-HEX": :
 :
360 PRINT :"IHRE EINGABE:"
370 CALL HCHAR(12,2,64,26)
380 CALL HCHAR(20,2,64,26)
390 CALL VCHAR(12,2,64,9)
400 CALL VCHAR(12,28,64,9)
410 FOR I=2 TO 8
420 CALL COLOR(I,2,4)
430 NEXT I
```

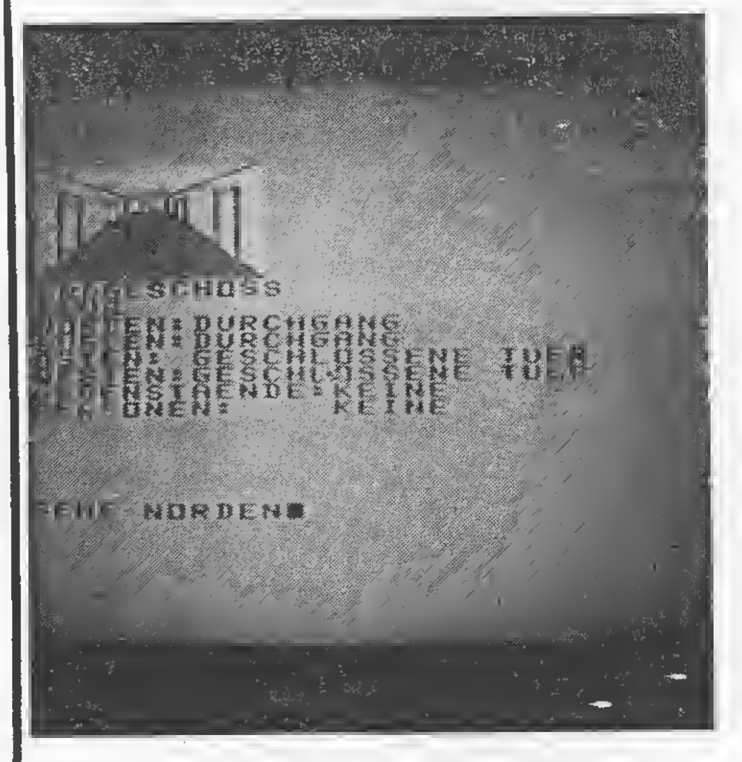
L I S T I N G S

```
440 CALL SCREEN(4)
450 RESTORE
460 CALL KEY(3,X,Y)
470 CALL SOUND(-9,-1,25)
480 IF (X<49)+(X>54)THEN 460
490 CALL CLEAR
500 ON X-48 GOTO 540,760,900
,1050,1220,1360
510 END
520 REM *DEZ-BIN*
530 REM
540 INPUT "DIE DEZIMALZAHL:"
:DZ
550 IF DZ<>INT(DZ)THEN 540
560 IF DZ>268435455 THEN 540
570 IF DZ>=0 THEN 610
580 PRINT : :"DA IHRE ZAHL N
EGATIV IST,...HABE ICH DEN W
ERT 65536.....DAZUADDIERT,UM
 DAS KOMPLE-..MENT ZU ERHALT
EN."
590 DZ=DZ+65536
600 GOTO 570
610 Z=1
620 B$=""
630 FOR C=27 TO 0 STEP -1
640 Z=2^C
650 IF DZ<Z THEN 690
660 B$=B$&"1"
670 DZ=DZ-Z
680 GOTO 700
690 B$=B$&"0"
700 NEXT C
710 PRINT : :"UND HIER IST D
IE BINAERZAHL:": :B$
720 GOTO 1550
730 REM
740 REM *BIN-DEZ*
750 REM
760 INPUT "DIE BINAERZAHL:..
............................
...........":B$
770 IF LEN(B$)>28 THEN 760
780 DZ=0
790 Z=LEN(B$)
800 FOR C=0 TO Z-1
810 X$=SEG$(B$,Z-C,1)
820 IF (X$="0")+(X$="1")THEN
 830 ELSE 1610
830 DZ=DZ-(X$="1")*2^C
840 NEXT C
850 PRINT : :"UND HIER DIE D
EZIMALZAHL:": :DZ: :
860 GOTO 1550
870 REM
880 REM *HEX-BCD*
890 REM
900 INPUT "DIE HEX-ZAHL:....
............................
..........":H$
910 IF LEN(H$)>28 THEN 900
920 PRINT : :"UND HIER DIE B
CD-ZAHL:": :
930 FOR C=1 TO LEN(H$)
940 X$=SEG$(H$,C,1)
950 X=POS(G$,X$,1)
960 IF X=0 THEN 1610
970 PRINT F$(X-1);" ";
980 NEXT C
990 GOTO 1550
1000 DATA 0000,0001,0010,001
1,0100,0101,0110,0111,1000
1010 DATA 1001,1010,1011,110
0,1101,1110,1111
1020 REM
1030 REM *BCD-HEX*
1040 REM
1050 INPUT "IHRE BCD-ZAHL(OH
NE ZWISCHEN-ABSTAENDE):.....
............................
...........":BC$
1060 H$=""
1070 X=LEN(BC$)
1080 IF X/4<>INT(X/4)THEN 10
50
1090 FOR I=1 TO X STEP 4
1100 X$=SEG$(BC$,I,4)
1110 FOR J=0 TO 15
1120 IF F$(J)=X$ THEN 1150
1130 NEXT J
1140 GOTO 1610
1150 H$=H$&SEG$(G$,J+1,1)
1160 NEXT I
1170 PRINT : :"UND HIER DIE
HEX-ZAHL:": :H$
1180 GOTO 1550
1190 REM
1200 REM *HEX-DEZ*
1210 REM
1220 INPUT "DIE HEX-ZAHL:...
............................
...........":H$
1230 Z=LEN(H$)
1240 DZ=0
1250 FOR I=1 TO Z
1260 X$=SEG$(H$,I,1)
1270 X=POS(G$,X$,1)
1280 IF X=0 THEN 1610
1290 DZ=DZ+(X-1)*2^((Z-I)*4)
1300 NEXT I
1310 PRINT : :"UND HIER DIE
DEZIMALZAHL:": :DZ
1320 GOTO 1550
1330 REM
1340 REM *DEZ-HEX*
1350 REM
1360 INPUT "IHRE DEZIMALZAHL
:...........................
...........":DZ
1370 IF DZ>268435455 THEN 13
60
1380 IF DZ<>INT(DZ)THEN 1360
1390 IF DZ>=0 THEN 1430
1400 PRINT : :"DA IHRE ZAHL
NEGATIV IST,...HABE ICH DEN
WERT 65536.....DAZUADDIERT,U
```

```
M DAS KOMPLE-..MENT ZU ERHAL
TEN."
1410 DZ=DZ+65536
1420 GOTO 1390
1430 H$=""
1440 FOR I=24 TO 0 STEP -4
1450 Z=2^I
1460 X=INT(DZ/Z)
1470 H$=H$&SEG$(G$,X+1,1)
1480 DZ=DZ-X*Z
1490 NEXT I
1500 PRINT : :"HIER DIE HEX-
ZAHL:": :H$
1510 GOTO 1550
1520 REM
1530 REM *WARTESCHLEIFE*
1540 REM
1550 PRINT : :"TASTE-->ZURUE
CK"
1560 CALL KEY(3,X,Y)
1570 IF Y=0 THEN 1560 ELSE 2
70
1580 REM
1590 REM *ERROR*
1600 REM
1610 CALL SOUND(100,-7,0)
1620 PRINT : :"*ERROR*": :"F
EHLERHAFTE EINGABE !!": :
1630 GOTO 1550
```


SCHULHAUS

Ein Schüler hat irgendwo im Schulhaus seinen Schulranzen verloren und Sie müssen diesen nun wiederfinden.
Während Sie durch das Gebäude gehen, treffen Sie andere Schüler, Lehrer und Hausmeister, die Ihnen wertvolle Tips geben können, allerdings manchmal etwas empfindlich reagieren oder nichts sagen wollen. Ferner liegen mehrere Gegenstände verstreut, die man mitnehmen kann. Es ist jedoch nicht möglich, mehr als vier Gegenstände gleichzeitig mit sich zu tragen. Besonders wichtig ist es, einen Dietrich bei sich zu haben, da man ohne ihn den Ranzen nicht finden kann. Wenn



Textadventure

Sie innerhalb des Zeitlimits den Ranzen finden und an den Ausgangspunkt zurückkehren, haben Sie die Aufgabe gelöst.
Wenn nicht, können Sie die Suche noch einmal von vorn beginnen.
Sie steuern den Schüler durch Zwei- und Drei-Wort-Befehle mit dem Aufbau Verb-(Präposition)-Substantiv. Die Wörter, die der Computer versteht, sind in den Programmzeilen 370 bis 400 zu finden.
Das Programm war anfänglich so lang, daß es die Speicherkapazität überschritt. Daher habe ich die Zeichendefinierung ausgelagert. Diese bildet nun ein Vorprogramm, von dem aus das Hauptprogramm nachgeladen wird. Deshalb muß beim Ausprobieren des Vorprogrammes in Zeile 300 der Befehl "RUN CS1" weggelassen werden. Vor dem Abspeichern muß er aber unbedingt wieder hinzugefügt werden, da sonst das Hauptprogramm nicht mehr nachgeladen werden kann.

Wird das Hauptprogramm nicht vom Vorprogramm, sondern manuell durch Eingabe des Befehls "RUN" gestartet, wird anstatt der Grafik ein Durch-

ANZEIGENSERVICE

Die große Börse für jeden Zweck in der TI REVUE. Kostenlos für Privat-Inserenten. Sportbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post – Freimachen nicht vergessen! – Unsere Adresse steht auf dem Coupon, ebenso der Preis für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht TI-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht angenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.

Name ______________________

Vorname ______________________

Straße/Hausnr. ______________________

PLZ/Ort ______________________

ABO-SERVICE-KARTE

COUPON

Ja, ich möchte von Ihrem Angebot Gebrauch machen.

Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten zwölf Ausgaben an untenstehende Anschrift. Sollte ich nicht vier Wochen vor Ablauf schriftlich kündigen, läuft diese Abmachung automatisch weiter.

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

Name ______________________

Vorname ______________________

Straße/Hausnr. ______________________

PLZ/Ort ______________________

Ich bezahle:

- ☐ per beiliegendem Verrechnungsscheck
- ☐ gegen Rechnung
- ☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort ______________________

Kontonummer ______________________

Bankleitzahl ______________________
(steht auf jedem Kontoauszug)

Unterschrift ______________________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift

TI REVUE
Abo-Service 6/86
Postfach 1107
8044 UNTERSCHLEISSHEIM

PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen
die Listings dieses Heftes auf
☐ Diskette zum Preis von (25,– DM)
☐ Kassette zum Preis von (10,– DM)

Ich zahle:
Bar – per beigefügtem Geld ( )
per beigefügtem Scheck ( )
Gegen Bankabbuchung am Versandtag ( )
Zutreffendes bitte ankreuzen!

Meine Bank (mit Ortsname) ....

Meine Kontonummer ....

Meine Bankleitzahl .... (steht auf jedem Bankauszug)

Vorname .... Nachname ....

Str./Nr. .... PLZ / Ort ....

Hiermit bestätige ich mit meiner Unterschrift, Ihre Verkaufsbedingungen gelesen zu haben und zu akzeptieren.

Unterschrift ....

TI-REVUE
KASSETTENSERVICE 6/86
Postfach 1107
8044 Unterschleißheim

Verkaufsbedingungen: Versand nur gegen Vorauskasse oder Bankabbuchung.
Umtauschrecht bei Nichtfunktionieren. Keine Nachnahme.

RESERVIERUNGS-SERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen

1 Exemplar TI SPECIAL (Nr. 4) 14,80 .... Exemplar(e) TI ASSEMBLER SPECIAL 19,80

Zutreffendes bitte ankreuzen!
Ich zahle:
per beigefügtem Scheck / Schein ( )
Gegen Bankabbuchung am Versandtag ( )
Meine Bank (mit Ortsname) .... (4)

Meine Kontonummer ....

Meine Bankleitzahl .... (steht auf jedem Bankauszug)

Vorname .... Nachname ....

Str./Nr. .... PLZ / Ort ....

Verkaufsbedingungen: Versand nur gegen Vorkasse oder Bankabbuchung. Keine Nachnahme!

Unterschrift ....

Bitte ausschneiden und einsenden an
TI-REVUE
Special-Service 6/86
Postfach 1107
8044 Unterschleißheim

VERDIENEN SIE GELD MIT IHREM COMPUTER!

Haben Sie einen TI 99/A? Können Sie programmieren? In Basic oder Maschinensprache? Dann bietet TI – REVUE Ihnen die Möglichkeit, damit Geld zu verdienen.

Wie? Ganz einfach. Sie senden uns die Programme, die Sie für einen Abdruck als geeignet halten, zusammen mit einer Kurzbeschreibung, aus der auch die verwendete Hardware – eventuelle Erweiterungen – benutzte Peripherie – hervorgehen muß (Schauen Sie sich dazu den Kopf unserer Programmlistings an.)

Benötigt werden: Zwei Listings des Programms sowie eine Datenkassette oder Diskette! Wenn die Redaktion sich überzeugt hat, daß dieses Programm läuft und sich zum Abdruck eignet, zahlen wir Ihnen pro Programm je nach Umfang bis zu DM 300,–! Sollten Sie keinen Drucker haben, genügt der Datenträger.

Sie erhalten Ihre Kassette/Diskette selbstverständlich zurück, wenn Sie einen ausreichend frankierten Rückumschlag mit Ihrer Adresse beifügen.

Bei der Einsendung müssen Sie mit Ihrer Unterschrift garantieren, daß Sie der alleinige Inhaber der Urheber-Rechte sind! Benutzen Sie bitte anhängendes Formular! (Wir weisen darauf hin, daß auch die Redaktion amerikanische und englische Fachzeitschriften liest und „umgestaltete" Programme ziemlich schnell erkennt).

Um Ihnen die Arbeit zu erleichtern, finden Sie hier ein Formular. Sie können es ausschneiden oder fotokopieren.

PROGRAMMANGEBOT

Name des Einsenders: ______________________

Straße/Hausnr./Tel.: ______________________

Plz/Ort: ______________________

Hiermit biete ich Ihnen zum Abdruck folgende(s) Programm(e) an:

____

____

____

____

Benötigte Geräte: ______________________

____

Beigefügt ( ) Listings ( ) Kassette ( ) Diskette

Ich versichere, der alleinige Urheber des Programmes zu sein!
Hiermit ermächtige ich die Redaktion, dieses Programm abzudrucken und wirtschaftlich zu verwerten. Sollte es in den Kassetten-Service aufgenommen werden, erhalte ich auch dafür eine entsprechende Vergütung, das Copyright geht auf den Verlag über.

____
Rechtsverbindliche Unterschrift

TI-REVUE
Postfach 1107
8044 Lohhof

```
10 ! **********************
11 ! *     SCHULHAUS1      *
12 ! *      (Teil 1)       *
13 ! *                     *
14 ! *    Copyright by     *
15 ! *                     *
16 ! *  Christoph Heuer    *
17 ! *                     *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole   *
21 ! *     Ext. Basic      *
22 ! *    Cassettenrec.    *
23 ! *(oder Disk+32K-Erw.)*
24 ! *                     *
26 ! *  Speicherbelegung   *
27 ! *     2602 Bytes      *
28 ! *                     *
29 ! **********************
100 CALL CHAR(132,"00FF")::
CALL CLEAR :: CALL SCREEN(11
):: DISPLAY AT(4,6):"S C H U
 L H A U S":".....DDDDDDDDDD
DDDDDDD"
110 DISPLAY AT(7,9):"VON C.H
EUER" :: DISPLAY AT(10,9):"V
ORPROGRAMM"
120 FOR I=1 TO 14 :: READ L,
T1,T2,T3 :: CALL SOUND(L,T1,
1,T2,1,T3,1):: NEXT I
130 DATA 325,523,392,330,275
,392,330,262,400,392,330,262
,400,523,392,330,325,440,349
,262
140 DATA 275,440,349,262,600
,440,349,262,325,494,392,294
,275,523,392,294,400,587,494
,392
150 DATA 400,494,392,294,275
,523,392,330,310,659,523,392
,800,523,392,330
160 DATA 40,C0300C03,41,0000
0000C0300C03,42,0,43,55AA55A
A55AA55AA,132,00FF,45,000000
00FFAA55AA,134,1010101010101
01,47,0808080808080808
170 DATA 135,818181818181818
1,136,00000000C0B88781,137,0
000000031DE181,138,3F202020
2020202,57,F808080808080808,
139,20202020202020202
180 DATA 140,080808080808080
8,141,81C161B159AD57AB,142,2
0202020FFAA55AA,143,08080808
FFAA55AA,58,0102050A152A55AA
,59,8182858A95AAD5AA
190 DATA 88,80C0608058AC56AB
,60,C03F10101010101,61,03FC0
80808080808,62,101010101F2A5
5AA,63,08080808F8AC56AB,81,0
000303000303
200 DATA 89,00FF000000FF8181
,133,00FF0000000F0808,91,00F
F000000F0101,92,8181FF818181
8181,93,08080F0808080808,94,
1010F0101010101
210 DATA 95,8181818181FF,96,
08080808080F,97,1010101010F,
98,000000000030C30C,99,030C30
C,100,00302C2424242424,101,2
526252A352A55AA
220 DATA 102,000C34242424242
4,103,A4E464B45CAC56AB,104,F
F808382828282FF,105,FF01C141
414141FF,106,704C45454545454
5,107,0000C03017141414
230 DATA 108,C03F101010D0505
,109,45457D4545454545,110,14
14F71414141414,111,5050D0505
050505,112,4545454545454545,
113,1414141414141415
240 DATA 114,505050505050901
,115,4545454545454544,116,16
141020408,117,4850604,118,03
FC080808B0A0A,119,0000030CE
8282828,128,000F30C
250 DATA 120,0E32A2A2A2A2A2A
2,121,0A0A0B0A0A0A0A0A,122,2
828EF2828282828,123,A2A2BEA2
A2A2A2A2,124,0A0A0A0A0A0A090
8,125,28282828282828A8
260 DATA 126,A2A2A2A2A2A2A2A
2,129,6828080402 01,130,A2A2A
2A2A2A2A222,131,120A0602,44,
0102050AF5AA55AA,127,00F00C0
3,56,80C060B05FAA55AA
270 DATA 48,0000000205020502
,49,050201,50,000000A050A050
A,51,50A04,37,1F3F373737,38,
FCFEF6F6F6,39,37373737370707
07,46,F6F6F6F6F6F0707
280 DATA 52,0707070707070707
,53,7070707070707070
290 FOR I=1 TO 81 :: READ CH
,CH$ :: CALL CHAR(CH,CH$)::
NEXT I
300 CALL CLEAR :: CALL CHAR(
86,"0044442810101010"):: RUN
 "CS1"
```

Achtung! Beim Eintippen des Programms ist zu beachten, daß die unterstrichenen Zeichen der folgenden Programmzeilen zusammen mit der CTRL-Taste einzugeben sind:
Teil 1: Zeile 100
Teil 2: Zeilen 110 und 1100–1190

LISTINGS

```
10 ! *********************
11 ! *     SCHULHAUS2      *
12 ! *      (Teil 2)       *
13 ! *                     *
14 ! *    Copyright by     *
15 ! *                     *
16 ! *  Christoph Heuer    *
17 ! *                     *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole   *
21 ! *     Ext. Basic      *
22 ! *    Cassettenrec.    *
23 ! *(oder Disk+32K-Erw.)*
24 ! *                     *
26 ! *  Speicherbelegung   *
27 ! *    11372 Bytes      *
28 ! *                     *
29 ! *********************
100 OPTION BASE 1
110 CALL CLEAR :: CALL SCREE
N(11):: DISPLAY AT(6,6):"S C
 H U L H A U S":".....DDDDDD
DDDDDDDDDD":"":"":"...BITTE
 HABEN SIE GEDULD"
120 DIM G(7,4),P(7,4),G$(8),
R(6),N$(6),N(6),B$(3),X$(3),
CC$(7),Q$(9),K$(8),CA(7),CB(
7),M$(2)
130 M$(1)="DER JUNGE" :: M$(
2)="DAS MAEDCHEN" :: FOR I=1
 TO 9 :: READ Q$(I):: NEXT I
 :: FOR I=1 TO 8 :: READ K$(
I):: NEXT I :: FOR I=4 TO 7
140 READ CC$(I):: NEXT I ::
FOR I=1 TO 7 :: READ CA(I),C
B(I):: NEXT I :: CC$(1)=CC$(
4):: CC$(2)=CC$(5):: CC$(3)=
CC$(6):: B=8 :: N(5),N(6)=4
150 G$(8)="KEINE" :: P$="KEI
NE" :: B$(1),X$(1)="ERDGESCH
OSS" :: B$(2)="ERSTEN STOCK"
 :: X$(2)="ERSTER STOCK" ::
B$(3)="ZWEITEN STOCK" :: X$(
3)="ZWEITER STOCK"
160 DATA FUENFTE,SECHSTE,SIE
BTE,ACHTE,NEUNTE,ZEHNTE,ELFT
E,ZWOELFTE,DREIZEHNTE
170 DATA 08083828383838,3C24
3C243C3C3C,02FE82FE,7E427E,0
2FDA28,1F387C7C7C38,3828FEBA
FEFE,0
180 DATA 00000304080A0808020
1000000000000000E0100828080
820C,0000000080A08000300000
00000000000000000008280800E
190 DATA 00000301000200000030
0000000000000000E0C00020000
0E,00010304080A08082A1100000
000000000C0E01008280808 2AC4
200 DATA 2,13,2,2,6,6,2,13,2
,2,6,6,2,8
210 DATA LEHRERZIMMER,2253,L
EHRERZIMMER,5244,REKTOR GVMN
ASIUM,4244,SEKRETARIAT,4214,
TREPPE,1215,REKTOR FOERDERST
UFE,1244,SEKRETARIAT
220 DATA 4214,TOILETTE,1144,
TOILETTE,4111,ELTERNSPRECHZI
MMER,2341,FLUR,4455,FLUR,545
4,FLUR,5454,FLUR,5555,FLUR,5
454,FLUR,5454
230 DATA FLUR,5445,MUSIKRAUM
,4124,AUFENTHALTSRAUM,2141,F
LUR,4515,SEKRETARIAT,1444,RE
KTOR HAUPTSCHULE,4412,EINGAN
G,1516
240 DATA REKTOR GESAMTSCHULE
,1442,SEKRETARIAT,4411,FLUR,
1545,MUSIKRAUM,4423,HAUSMEIS
TER,2141,FLUR,4545,REKTOR RE
ALSCHULE,4421
250 DATA SCHULHOF,2255,SCHUL
HOF,5655,SCHULHOF,5225,MUSIK
SAMMLUNG,2131,FLUR,3535,MUSI
KRAUM,3321,TREPPE,2151,FLUR,
5555,EINGANG
260 DATA 5161,SCHULHOF,6555,
SCHULHOF,5555,SCHULHOF,5565,
EINGANG,6151,FLUR,5555,TREPP
E,5121,1.,2131,FLUR,3545,1.,
4121,SCHULHOF
270 DATA 2552,SCHULHOF,5556,
SCHULHOF,5522,2.,2131,FLUR,3
535,LEERER RAUM,3121,1.,2141
,FLUR,4535,1.,3111,1.,1213,E
INGANG,1615
280 DATA 2.,1214,2.,1131,FLU
R,3545,2.,4121,TOILETTE,1144
,FLUR,4554,FLUR,5153,FLUR,53
54,FLUR,5555,FLUR,5453,FLUR,
5154
290 DATA FLUR,5534,2.,3121,T
OILETTE,1411,1.,1412,1.,1312
,1.,1412,TREPPE,1512,2.,1312
,2.,1412,TOILETTE,1441,TOILE
TTE,4111
300 DATA LEERER RAUM,2133,4.
,3214,4.,1214,4.,1213,TREPPE
,1215,4.,1214,4.,1213,TOILET
TE,1144,TOILETTE,4111
310 DATA 3.,2341,FLUR,4455,F
LUR,5451,FLUR,5354,FLUR,5553
,FLUR,5453,FLUR,5351,FLUR,54
35,4.,4121
320 DATA 3.,2141,FLUR,4535,5
.,3111,5.,1412,KAKAOSTAND,13
12,5.,1312,5.,1141,FLUR,4535
,4.,3121
330 DATA LEERER RAUM,2141,FL
UR,4545,5.,4121,,0,,0,,0,5.,
2131,FLUR,3545,4.,4121
340 DATA TREPPE,2151,FLUR,55
35,5.,3121,,0,,0,,0,5.,2141,
FLUR,4555,TREPPE,5121
350 DATA 6.,2141,FLUR,4535,5
.,3121,,0,,0,,0,3.,2141,FLUR
```

LISTINGS

```
,4545,3.,4121
360 DATA 6.,2131,FLUR,3545,5
.,4111,4.,1213,4.,1214,4.,12
14,4.,1131,FLUR,3545,3.;4121
370 DATA TOILETTE,1144,FLUR,
4554,FLUR,5153,FLUR,5354,FLU
R,5455,FLUR,5454,FLUR,5154,F
LUR,5544,3.,4121
380 DATA TOILETTE,1411,6.,14
12,6.,1312,LEERER RAUM,1412,
TREPPE,1512,3.,1412,3.,1412,
TOILETTE,1441,TOILETTE,4111
390 DATA LEERER RAUM,2244,6.
,4214,6.,1213,6.,1214,TREPPE
,1215,7.,1214,7.,1213,7.,123
3,LEERER RAUM,3223,BIOLOGIER
AUM,2431
400 DATA FLUR,3455,FLUR,5351
,FLUR,5454,FLUR,5554,FLUR,54
53,FLUR,5351,FLUR,5345,8.,43
21,BIOLOGIERAUM,2141,FLUR,45
35
410 DATA PHVSIKSAMMLUNG,3144
420 DATA PHVSIKRAUM,4412,FIL
MSAAL,1412,7.,1312,7.,1141,F
LUR,4535,8.,3121,BIOLOGIERAU
M,2141,FLUR,4545,PHVSIKRAUM,
4421
430 DATA ,0,,0,,0,8.,2131,FL
UR,3535,8.,3121,TREPPE,2151,
FLUR,5545,LEERER RAUM,4121,,
0,,0,,0,8.,2141,FLUR,4555
440 DATA TREPPE,5121,CHEMIER
AUM,2144,FLUR,4545,PHVSIKRAU
M,4124,,0,,0,,0,KUNSTRAUM,21
44,FLUR,4545,8.,4121,CHEMIES
AMMLUNG,2434
450 DATA FLUR,3535,PHVSIKSAM
MLUNG,3431,PHVSIKRAUM,3213,F
ILMSAAL,1214,KUNSTRAUM,1214,
KUNSTRAUM,1441,FLUR,4535,9.,
3121
460 DATA CHEMIERAUM,2444,FLU
R,4553,FLUR,5154,FLUR,5353,F
LUR,5455,FLUR,5453,FLUR,5154
,FLUR,5544,9.,4121,CHEMIESAM
MLUNG,2442
470 DATA CHEMIERAUM,4312,9.,
1412,9.,1312,TREPPE,1512,9.,
1312,9.,1412,TOILETTE,1441,T
OILETTE,4111
480 DATA ZIGARETTEN,COLABUEC
HSE,STUECK KUCHEN,PAUSENBROT
,DIETRICH,SPORTBEUTEL,SCHULR
ANZEN
490 DATA WAND,FENSTER,GESCHL
OSSENE TUER,OFFENE TUER,DURC
HGANG,EINGANG
500 DATA JUNGE,LEHRER,HAUSME
ISTER,JUNGE,LEHRER,HAUSMEIST
ER,MAEDCHEN
510 DATA WESTEN,WEST,W,NORDE
N,NORD,N,OSTEN,OST,O,SUEDEN,
SUED,S,OBEN,RAUF,R,UNTEN,RUN
TER,U
520 DATA GEHE,GEH,NIMM,GREIF
E,VERLIERE,GIB,FRAGE,BEFRAGE
530 FOR I=1 TO 7 :: FOR J=1
TO 4 :: RANDOMIZE :: A=INT(R
ND*3)+1
540 RANDOMIZE :: G(I,J)=INT(
RND*81)+1 :: FOR L=30 TO 50
STEP 9 :: IF L<G(I,J)AND G(I
,J)<L+4 THEN 540
550 NEXT L :: G(I,J)=G(I,J)+
A*100 :: IF I=7 AND J=1 THEN
 O=G(I,J):: ON A GOSUB 780,7
90,800 ELSE 580
560 FOR L=1 TO 4 :: IF VAL(S
EG$(STR$(W),L,1))>3 THEN 540
570 NEXT L
580 RANDOMIZE :: A=INT(RND*3
)+1 :: RANDOMIZE :: P(I,J)=I
NT(RND*81)+1+A*100
590 IF P(I,J)=141 THEN 580
600 NEXT J :: NEXT I :: R(1)
=-1 :: R(2)=-9 :: R(3)=1 ::
R(4)=9 :: R(5)=100 :: R(6)=-
100 :: G(6,2),G(6,3),G(6,4),
G(7,2),G(7,3),G(7,4)=0
610 O=141 :: RESTORE 480 ::
FOR I=1 TO 7 :: READ G$(I) :
: NEXT I :: FOR I=1 TO 6:: R
EAD N$(I):: NEXT I
620 B=8 :: ON INT(O/100)GOSU
B 780,790,800 :: IF SEG$(O$,
LEN(O$),1)="." THEN O$=O$(VA
L(SEG$(O$,1,1)))&" KLASSE"
630 P$="KEINE" :: FOR I=1 TO
 7 :: FOR J=1 TO 4 :: IF G(I
,J)=O THEN B=I
640 IF P(I,J)=O THEN GOSUB 8
10
650 NEXT J :: NEXT I :: CALL
 CLEAR :: CALL DELSPRITE(ALL
):: FOR I=1 TO 4 :: N(I)=VAL
(SEG$(STR$(W),I,1)):: NEXT I
 :: GOSUB 1090 :: GOSUB 1480
660 GOSUB 1500 :: DISPLAY AT
(8,1):X$(INT(O/100)):O$:"NOR
DENQ";N$(N(2)):"SUEDENQ";N$(
N(4)):"OSTENQ ";N$(N(3)):"WE
STENQ";N$(N(1))
670 DISPLAY AT(14,1):"GEGENS
TAENDEQ";G$(B):"PERSONENQ...
.";P$ :: FOR I=1 TO 7 :: FOR
 J=1 TO 4 :: IF G(I,J)=400 T
HEN Z=Z+1 :: DISPLAY AT(Z,12
):G$(I)
680 NEXT J :: NEXT I :: Z=0
:: IF O=141 AND G(7,1)=400 T
HEN 760 ELSE GOSUB 690 :: GO
TO 730
690 M=M+1 :: ACCEPT AT(21,1)
:E$ :: FOR I=1 TO LEN(E$)::
IF SEG$(E$,I,1)=" " THEN A$=
```

```
SEG$(E$,1,I-1):: F$=SEG$(E$,
I+1,LEN(E$)-I):: I=LEN(E$)
700 NEXT I :: RESTORE 520 ::
 FOR I=1 TO 4 :: FOR J=1 TO
2 :: READ L$ :: IF L$=A$ THE
N C=I :: J=2 :: I=4
710 NEXT J :: NEXT I :: IF S
EG$(F$,1,4)="NACH" THEN D$=S
EG$(F$,6,LEN(F$)-5) ELSED$=F
$
720 IF C=0 THEN DISPLAY AT(1
8,1):"DAS WORT ";A$:"KENNE I
CH NICHT" :: GOTO 620 ELSE R
ETURN
730 IF (E=2 OR E=3)AND(C=1 O
R C=4)THEN DISPLAY AT(18,1):
"DER ";P$;" SCHICKT DICH AUF
 DEN HOF" :: O=141 :: GOTO 7
50
740 ON C GOSUB 820,850,870,8
90
750 E=0 :: C=0 :: IF M>160 T
HEN DISPLAY AT(18,1):"BEDAUE
RE":"DIE ZEIT IST ABGELAUFEN
" :: GOTO 770 ELSE 620
760 DISPLAY AT(18,1):"GRATUL
IERE":"DU HAST ES GESCHAFFT"
770 DISPLAY AT(20,1):"WILLST
 DU NOCHEINMAL SUCHEN J ODER
 N" :: CALL KEY(0,K1,S1):: I
F S1=0 THEN 770 ELSE IF K1=7
4 THEN RUN ELSE END
780 RESTORE 210 :: FOR II=10
1 TO O :: READ O$,W :: NEXT
II :: RETURN
790 RESTORE 300 :: FOR II=20
1 TO O :: READ O$,W :: NEXT
II :: RETURN
800 RESTORE 390 :: FOR II=30
1 TO O :: READ O$,W :: NEXT
II :: RETURN
810 RESTORE 500 :: FOR L=1 T
O I :: READ P$ :: E=L :: NEX
T L :: RETURN
820 RESTORE 510 :: FOR I=1 T
O 6 :: FOR J=1 TO 3 :: READ
L$
830 IF L$=D$ AND(N(I)>3 OR(N
(I)=3 AND (G(5,1)=400 OR G(5
,2)=400 OR G(5,3)=400 ORG(5,
4)=400)))THEN V=I :: I=6 ::
J=3 ELSE IF I=6 THEN 1080
840 NEXT J :: NEXT I :: IF (
V=5 AND O>300)OR(V=6 AND O<2
00)OR((V=5 OR V=6)AND O$<>"T
REPPE")THEN 1080 ELSE O=O+R(
V):: RETURN
850 FOR I=1 TO 7 :: FOR J=1
TO 4 :: IF G(I,J)=O THEN IF
G$(I)=F$ AND H<4 THEN G(I,J)
=400 :: H=H+1 :: I=7 :: J=4
ELSE 1080 ELSE IF I=7 THEN 1
080
860 NEXT J :: NEXT I :: RETU
RN
870 FOR I=1 TO 7 :: FOR J=1
TO 4 :: IF G$(I)<>F$ THEN 88
0 ELSE IF G(I,J)=400 THEN G(
I,J)=O :: H=H-1 :: J=4 ELSE
IF J=4 THEN 1080
880 NEXT J :: NEXT I :: RETU
RN
890 IF SEG$(F$,1,4)<>"NACH"
THEN 1040 ELSE IF E=0 THEN 1
080
900 FOR I=1 TO 7 :: IF G$(I)
=D$ THEN U=I :: I=7 ELSE IF
I=7 THEN DISPLAY AT(18,1):".
..DANACH KANNST DU NICHT....
..FRAGEN"
910 NEXT I :: IF (E=7 OR E=4
)AND U=7 THEN DISPLAY AT(18,
1):M$(INT(E-1)/3)&"WEISS ES
NICHT" :: RETURN
920 IF E=1 THEN 980 ELSE IF
E=5 OR E=6 AND U<6 THEN DISP
LAY AT(18,1):"DER ";P$;"SCHI
CKT DICH AUF DEN HOF" :: O=1
41 :: RETURN
930 FOR J=1 TO 4 :: IF G(U,J
)<>400 AND G(U,J)<>0 THEN F=
VAL(SEG$(STR$(G(U,J)),1,1)):
: DISPLAY AT(18,1):"IM ";B$(
F):: Y1=J :: J=4
940 NEXT J :: IF F=INT(O/100
)THEN ON F GOSUB 1050,1060,1
070 ELSE RETURN
950 IF SEG$(S$,2,1)="." THEN
 S$=Q$(VAL(SEG$(S$,1,1)))&"K
LASSE"
960 IF S$="" THEN S$="DEN HA
ST DU"
970 DISPLAY AT(19,1):S$ :: S
$="" :: RETURN
980 DISPLAY AT(18,1):"DER JU
NGE ANTWORTET NICHT ERWILL S
EINE ANTWORT BEZAHLT  HABEN"
990 Y=0 :: GOSUB 690 :: ON C
 GOTO 820,850,1000,890
1000 DISPLAY AT(18,1)SIZE(84
):"" :: FOR J=1 TO 7 :: IF G
$(J)=D$ THEN T=J :: J=7
1010 NEXT J :: FOR J=1 TO 4
:: IF G(T,J)=400 THEN G(T,J)
=300 ::  J=4 ELSE IF J=4THEN
 DISPLAY AT(18,1):"DAS HAST
DU NICHT" :: ACCEPT AT(21,1)
:E$ :: J=4 :: Y=1 :: GOTO 10
30
1020 IF T>2 THEN DISPLAY AT(
18,1):"DER JUNGE GIBT DIR TR
OTZDEM KEINE AUSKUNFT" :: J=
4 :: Y=1 ELSE 930
1030 NEXT J :: IF Y=1 THEN 9
80 ELSE 930
1040 DISPLAY AT(18,1):"WONAC
```

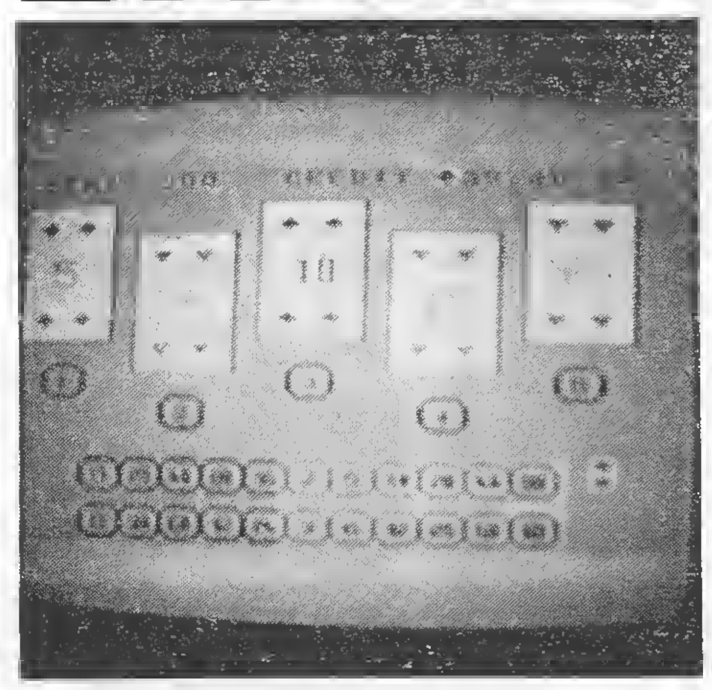
LISTINGS

```
H WILLST DU FRAGEN?" :: M=M-
1 :: GOTO 690
1050 RESTORE 210 :: FOR I=10
1 TO G(U,Y1):: READ S$,W$ ::
 NEXT I :: RETURN
1060 RESTORE 300 :: FOR I=20
1 TO G(U,Y1):: READ S$,W$ ::
 NEXT I :: RETURN
1070 RESTORE 390 :: FOR I=30
1 TO G(U,Y1):: READ S$,W$ ::
 NEXT I :: RETURN
1080 DISPLAY AT(18,1):"DAS G
EHT NICHT" :: RETURN
1090 IF O$="SCHULHOF" THEN G
OTO 1410 ELSE IF AA=1 THEN A
A=0 :: GOSUB 1420
1100 DATA ()******bc,**<DDDD
=**,**F****/**,**F****/**,**
>----?**,*:++++++X*,:+++++++
+X
1110 DATA 3,()*,jkl,mno,pqr,
st>,u:+,:++
1120 DATA 2,()*,*H<,*GF,*GF,
*G>,*;+,:++
1130 DATA 3,***,DDD,***,***,
---,+++,+++
1140 DATA <YE[Y=,F\]^\/,F_`a
_/,>----?
1150 DATA ()**bc,*dhif*/,*e+
+g*,:++++X
1160 DATA <DDDD=,F*J9*/,F*KL
*/,>-NO-?
1170 DATA 3,*bc,vwx,yz{,|}~,
?AB,+XC,++X
1180 DATA 2,bc,I*,G*,G*,G*,M
*,+X
1190 DATA 3,***,DDD,***,***,
---,+++,+++
1200 CALL COLOR(2,11,11,3,10
,15,4,11,11,8,11,11,9,11,11,
10,11,11,11,11,11,12,11,11,1
3,11,11,14,11,11)
1210 RESTORE 1090 :: FOR I=1
 TO 7 :: READ R$ :: DISPLAY
AT(I,1)SIZE(10):R$ :: NEXT I
1220 ON N(2)GOTO 1270,1230,1
240,1240,1250,1240
1230 RESTORE 1140 :: GOTO 12
60
1240 RESTORE 1160 :: GOTO 12
60
1250 RESTORE 1150
1260 FOR I=2 TO 5 :: READ R$
 :: DISPLAY AT(I,3)SIZE(6):R
$ :: NEXT I
1270 ON N(3)GOTO 1320,1280,1
290,1290,1300,1290
1280 RESTORE 1170 :: GOTO 13
10
1290 RESTORE 1180 :: GOTO 13
10
1300 RESTORE 1190
1310 READ L1 :: FOR I=1 TO 7
 :: READ R$ :: DISPLAY AT(I,
11-L1)SIZE(L1):R$ :: NEXT I
1320 ON N(1)GOTO 1370,1330,1
350,1350,1340,1350
1330 RESTORE 1110 :: GOTO 13
60
1340 RESTORE 1130 :: GOTO 13
60
1350 RESTORE 1120
1360 READ L1 :: FOR I=1 TO 7
 :: READ R$ :: DISPLAY AT(I,
1)SIZE(L1):R$ :: NEXT I
1370 CALL COLOR(2,2,15,4,2,1
5,8,2,15,9,2,15,10,2,15,11,2
,15,12,2,15,13,2,15,14,2,15)
1380 IF N(1)=5 AND N(2)=5 TH
EN CALL HCHAR(2,5,127):: CAL
L HCHAR(5,5,44)
1390 IF N(2)=5 AND N(3)=5 TH
EN CALL HCHAR(2,10,128):: CA
LL HCHAR(5,10,56)
1400 RETURN
1410 RESTORE 1430 :: AA=1 ::
 GOTO 1450
1420 RESTORE 1440
1430 DATA 0000666600666666,6
66666,0000070F1F1F1F1F,1F1F1
F1F1F1F1F1F,0000E0F0F8F8F8F8
,F8F8F8F8F8F8F8F8
1440 DATA FF808382828282FF,F
F01C141414141FF,704C45454545
4545,0000C03017141414,C03F10
1010D0505,45457D4545454545
1450 CALL COLOR(2,11,11,10,1
1,11,11,11,11):: FOR I=104 T
O 109 :: READ R$ :: CALL CHA
R(I,R$):: NEXT I :: IF O$<>"
SCHULHOF" THEN RETURN
1460 RESTORE 1470 :: FOR I=1
 TO 7 :: READ I$ :: DISPLAY
AT(I,1)SIZE(10):I$ :: NEXT I
 :: CALL COLOR(2,15,15,3,10,
15,10,2,7,11,2,7):: RETURN
1470 DATA hhhhhhhhh,iiiiii
iii,hhhhhhhhh,iiiiiiiii,hh
hhjlhhhh,iiiikmiiii,********
**
1480 CALL CHAR(55,K$(B)):: I
F B<8 THEN DISPLAY AT(7,4)SI
ZE(1):"7"
1490 RETURN
1500 IF E<1 THEN RETURN ELSE
 CALL VCHAR(4,7,42,4):: CALL
 VCHAR(4,8,42,4)
1510 CALL CHAR(33,CC$(E))::
DISPLAY AT(4,5)SIZE(2):"02"
:: DISPLAY AT(5,5)SIZE(2):"1
3" :: CALL SPRITE(#3,33,CA(E
),25,49,#4,34,2,33,49,#5,35,
CA(E),25,57)
1520 CC=CB(E):: CALL SPRITE(
#6,36,2,33,57,#7,37,CC,36,49
,#8,38,CC,36,57,#9,39,CC,41,
```

```
49,#10,46,CC,41,57)
1530 CALL SPRITE(#11,52,CC,4
9,49,#12,53,CC,49,57):: RETU
RN
```

LISTINGS

POKER-BANDIT



Das Spiel läuft nur mit Ext. Basic und benötigt ca. 11000 Byte. Ist das Spiel gestartet und sind alle Felder erstellt, bleiben noch ca. 100 Byte Speicherplatz.
Bei diesem Spiel handelt es sich um eine Kombination des altbekannten Pokerspiels mit einem Geldspielgerät. Nachdem das Spiel gestartet wurde, erscheint in der obersten Bildschirmzeile die Anzeige für Serie und Kredit. Darunter die fünf Spielkarten von der Rückseite gesehen. Unter den Karten fünf Felder, 1–5 numeriert. Als letztes folgen noch zwei Risikoleisten, eine von 0 – 90, die andere von 0 – 100.
Nachdem ein kurzer Ton gegeben worden ist, kann der gewünschte Einsatz eingegeben werden. Dafür drückt man die Tasten 1 – 9. Die Eingabe wird mit 0 beendet. Ein Spiel kostet 0.30 DM. Jetzt werden die fünf Karten aufgedeckt. Die Karten, die eine Gewinnkombination nach den Regeln des Pokerspiels ergeben, werden in dem entsprechenden Feld durch einen roten Punkt angezeigt. Diese Karten werden vom Computer automatisch gehalten. Will der Spieler noch andere Karten halten, so muß er die entsprechende Zahltaste drücken (1 – 5). Es erscheint dann der rote Punkt. Durch nochmaliges Drücken der Taste wird der Punkt wieder gelöscht. Die Karten, die gehalten werden, werden bei der zweiten Ziehung nicht nochmal ausgeteilt. Auch die Karten, die

Bitte lesen Sie weiter auf Seite 44

```
10 ! *********************
11 ! *                   *
12 ! *  POKERGELDSPIELER *
13 ! *                   *
14 ! *    Copyright by   *
15 ! *                   *
16 ! * Michael Lindenbaum *
17 ! *                   *
19 ! * Benoetigte Geraete *
20 ! *   TI99/4A  Konsole *
21 ! *      Ext. Basic    *
22 ! *    Cassettenrec.   *
23 ! *(oder DISK+32K-Erw.)*
24 ! *                   *
26 ! *  Speicherbelegung  *
27 ! *     10231 Bytes    *
28 ! *                   *
29 ! *********************
100 OPTION BASE 1
110 DIM BL(4,13),WE(5),FA(5)
,WEG(5),GE(20,2),TE$(20),CO(
4)
120 CRE,SER=0 :: CALL CLEAR
:: RANDOMIZE
130 CALL CHARDEF
140 CALL FARBE
150 CALL BILD1
160 CALL DATEN(GE(,),TE$(),C
O())
170 IF CRE<.3 THEN CALL CRED
IT(CRE)
180 IF SER>0 AND S1=0 THEN S
ER=SER-1 :: GOSUB 390
190 S1=0 :: IF CRE>=.3 THEN
210
200 CALL CLEAR :: CALL CHARS
ET :: CALL DELSPRITE(ALL)::
STOP
210 CALL MISCH(WE(),FA(),BL(
,),WEG())
220 CALL AUSTEI(1,5,4,WE(),F
A(),CO(),WEG())
230 CALL AUSTEI(2,4,6,WE(),F
A(),CO(),WEG())
240 CALL PAARE(WE(),WEG(),GE
W)
250 IF GEW=0 THEN CALL FLUSH
(FA(),WEG(),GEW):: CALL STRE
ET(WE(),WEG(),GEW)
260 CALL HALTEN(WEG()):: CRE
=CRE-.3 :: GOSUB 380
270 CALL ZWEZIE(WE(),FA(),BL
(,),WEG(),CO())
280 FOR I=1 TO 5 :: IF WEG(I
)=0 THEN 300
290 NEXT I :: GOTO 340
300 FOR I=1 TO 5 :: WEG(I)=0
 :: NEXT I
310 CALL PAARE(WE(),WEG(),GE
W):: IF GEW>0 THEN 340
320 CALL FLUSH(FA(),WEG(),GE
W):: CALL STREET(WE(),WEG(),
GEW)
```

LISTINGS

```
330 IF GEW=0 THEN 350
340 GOSUB 420 :: CALL WIN(GE
W,GE(,),WEG(),TE$(),CRE,SER,
S1)
350 FOR I=1 TO 5 :: CALL COL
OR(#I+5,1):: WEG(I)=0 :: NEX
T I :: DISPLAY AT(18,1):RPT$
("J",28)
360 CALL COLOR(#11,1):: CALL
 RUECK(1,5,4,WEG()):: CALL R
UECK(2,4,6,WEG())
370 GOTO 170
380 DISPLAY AT(1,20)SIZE(9):
USING "###.##JDM":CRE :: RET
URN
390 DISPLAY AT(1,7)SIZE(3):U
SING "###":SER :: RETURN
400 FOR I=333 TO 999 STEP 33
3 :: CALL SOUND(150,I,6,I+11
,7,I-11,8):: NEXT I :: RETUR
N
410 FOR I=666 TO 222 STEP -2
22 :: CALL SOUND(150,I,8,I-1
1,7,I+11,6):: NEXT I :: RETU
RN
420 FOR I=333 TO 666 STEP 33
3 :: CALL SOUND(150,I,5):: N
EXT I :: RETURN
430 SUB CHARDEF
440 CALL CHAR(96,"00F71515F5
8585F7",97,"00F71515751515F7
",98,"008785A5A5F52527",99,"
00F78585F59595F7")
450 CALL CHAR(100,"00F79595F
59595F7",101,"002F61A12F2828
2F",102,"002F68A82F29292F",1
03,"00E8282AEA8F82E2")
460 CALL CHAR(104,"002765A52
5252527",105,"00F71414F78181
F7",106,"00F78585F51515F7",1
07,"00F79595F51515F7",108,"0
05FD555555555F")
470 A$="0000000103060C18" ::
 CALL CHAR(112,A$,120,A$)::
A$="00000080C0603018" :: CAL
L CHAR(113,A$,121,A$)
480 A$=RPT$("18",8):: CALL C
HAR(114,A$,122,A$):: A$="000
000FFFF" :: CALL CHAR(115,A$
,123,A$)
490 A$="180C060301000000" ::
 CALL CHAR(116,A$,124,A$)::
A$="183060C080000000" :: CAL
L CHAR(117,A$,125,A$)
500 A$="00000081C3663C18" ::
 CALL CHAR(118,A$,126,A$)::
A$="183C66C381000000" :: CAL
L CHAR(119,A$,127,A$)
510 CALL CHAR(88,RPT$("7F",8
),89,RPT$("FE",8),90,"007F7F
7F7F7F7F7F",91,"00FEFEFEFEFE
FEFE")
520 CALL CHAR(92,RPT$("7F",7
),93,RPT$("FE",7),94,"00FFFF
FFFFFFFFFF",95,RPT$("FF",7))
530 CALL CHAR(128,"007E313B3
633337E",129,"007E3211191137
7E",130,"007B72363C363673",1
31,"003F666C746466EF")
540 CALL CHAR(132,"6CFEFEFE7
C383810",133,"10387CFEFE7C38
10",32,"10387CFEFED61038",33
,"10381054EE541028")
550 CALL CHAR(139,"C0E070381
C0E0703",136,"9E91909F81918F
E0",137,"007E7E7E7E7E7E7E",138
,"3C7EFFFFFFFF7E3C")
560 CALL CHAR(111,"007C04081
020407C",81,"004444281010101
0",74,"")
570 SUBEND
580 SUB FARBE
590 CALL MAGNIFY(2):: CALL S
CREEN(12)
600 CALL COLOR(0,16,16,1,2,1
6,8,16,2,12,13,12,13,9,16,14
,6,16)
610 SUBEND
620 SUB BILD1
630 CALL HCHAR(1,1,74,768)
640 DATA "SERIEJ000JJJCREDIT
J000.00JDM"
650 DATA "JJJJJJJJJJJJJJJJJJ
JJJJJJJJJJ"
660 DATA "^^^[JJJJJJJZ^^^[JJ
JJJJJZ^^^["
670 DATA "...YJJJJJJJX...YJJ
JJJJJX...Y"
680 DATA "...YJZ^^^[JX...YJZ
^^^[JX...Y"
690 DATA "...YJX...YJX...YJX
...YJX...Y"
700 DATA "...YJX...YJX...YJX
...YJX...Y"
710 DATA "...YJX...YJX...YJX
...YJX...Y"
720 DATA "...YJX...YJX...YJX
...YJX...Y"
730 DATA "...YJX...YJX...YJX
...YJX...Y"
740 DATA "___]JX...YJ\___]JX
...YJ\___]"
750 DATA "JJJJJX...YJJJJJJJX
...YJJJJJJ"
760 DATA "psqJJ\___]JJpsqJJ\
___]JJpsqJ"
770 DATA "r1rJJJJJJJJr3rJJJ
JJJJJJr5rJ"
780 DATA "tsuJJJpsqJJJtsuJJJ
psqJJJtsuJ"
790 DATA "JJJJJJr2rJJJJJJJJJ
r4rJJJJJJJ"
800 DATA "JJJJJJtsuJJJJJJJJJ
tsuJJJJJJJ"
810 DATA "JJJJJJJJJJJJJJJJJJ
JJJJJJJJJJ"
```

LISTINGS

```
820 DATA "JJpsvsvsvsvs~{~{~{
~{~{~{yJJJ"
830 DATA "JJr0r`rbrdrfz2z5zh
z`zbzkzJJJ"
840 DATA "JJtswswswsws.{.{.{
.{.{.{}JJJ"
850 DATA "JJpsvsvsvsvs~{~{~{
~{~{~{yJJJ"
860 DATA "JJr0rarcrergz3z6ze
zizjzIzJJJ"
870 DATA "JJtswswswsws.{.{.{
.{.{.{}JJJ"
880 RESTORE 640 :: FOR I=1 T
O 24 :: READ M$ :: DISPLAY A
T(I,1):M$ :: NEXT I
890 CALL HCHAR(3,2,90):: CAL
L VCHAR(4,2,88,7):: CALL HCH
AR(11,2,92)
900 CALL SPRITE(#1,136,2,44,
21,#2,136,2,60,69,#3,136,2,4
4,117,#4,136,2,60,165,#5,136
,2,44,213)
910 CALL SPRITE(#6,138,1,101
,21,#7,138,1,117,69,#8,138,1
,101,117,#9,138,1,117,165,#1
0,138,1,101,213)
920 CALL SPRITE(#11,138,1,14
9,37)
930 SUBEND
940 SUB DATEN(GE(,),TE$(),CO
())
950 DATA .2,53,.3,53,.4,69,.
6,69,.8,85,1.2,85,1.6,101,2.
4,101,2,117,3,117
960 DATA 5,133,6,133,10,149,
12,149,20,165,25,165,40,181,
50,181,90,197,100,197
970 DATA oWILLING,DOPPELoWIL
LING,DRILLING,FULLJHOUSE,STR
ASSE,FLUSH
980 DATA DRILLINGJAS,VIERLIN
G,VIERLINGJAS,STRAIGHTJFLUSH
,ROQALJFLUSH
990 DATA 132,133,32,33
1000 RESTORE 950 :: FOR I=1
TO 20 :: READ GE(I,1),GE(I,2
):: NEXT I
1010 RESTORE 970 :: READ TE$
(1):: READ TE$(2)
1020 FOR I=5 TO 8 :: READ TE
$(I):: NEXT I :: READ TE$(11
):: READ TE$(12):: READ TE$(
16):: READ TE$(18):: READ TE
$(20)
1030 RESTORE 990 :: FOR I=1
TO 4 :: READ CO(I):: NEXT I
1040 SUBEND
1050 SUB CREDIT(CRE)
1060 CALL SOUND(150,400,5)
1070 CALL KEY(0,K,S):: IF S=
0 OR K<48 OR K>57 THEN 1070
1080 IF K=48 THEN 1090 ELSE
CRE=CRE+(K-48):: GOSUB 380 :
: GOTO 1060
1090 SUBEND
1100 SUB MISCH(WE(),FA(),BL(
,),WEG())
1110 FOR I=1 TO 4 :: FOR J=1
 TO 13 :: BL(I,J)=1 :: NEXT
J :: NEXT I
1120 FOR I=1 TO 5
1130 CALL ZUF(A,B,BL(,))
1140 FA(I)=A :: WE(I)=B :: B
L(A,B)=0 :: NEXT I
1150 SUBEND
1160 SUB AUSTEI(A,B,Z,WE(),F
A(),CO(),WEG())
1170 FOR I=A TO B STEP 2 ::
IF WEG(I)=1 THEN 1220
1180 A=CO(FA(I)):: D=WE(I)::
 B=-((D<9)*(D+49)+(D=9)*(104
)+(D>9)*(D+118)):: C=-((FA(I
)<3)*(10)+(FA(I)>2)*(2))
1190 CALL SOUND(250,-4,0)::
S=-3+(I*6):: CALL COLOR(#I,1
):: FOR J=0 TO 2 :: CALL VCH
AR(Z,S+J,30,7):: NEXT J
1200 CALL HCHAR(Z,S,A):: CAL
L HCHAR(Z,S+2,A):: CALL HCHA
R(Z+6,S,A):: CALL HCHAR(Z+6,
S+2,A)
1210 CALL PATTERN(#I,B):: CA
LL COLOR(#I,C)
1220 NEXT I
1230 SUBEND
1240 SUB PAARE(WE(),WEG(),GE
W)
1250 PA=1 :: GEW,NO,PAR,DR=0
1260 FOR I=1 TO 4 :: FOR J=I
+1 TO 5
1270 IF WE(I)=WE(J)THEN PA=P
A+1 :: WEG(I),WEG(J)=1
1280 NEXT J
1290 IF PA=4 AND WE(I)=13 TH
EN GEW=16 :: GOTO 1370
1300 IF PA=4 THEN GEW=12 ::
GOTO 1370
1310 IF PA=3 AND WE(I)=13 TH
EN DR=5 :: NO=WE(I):: GOTO 1
360
1320 IF PA=3 THEN DR=1 :: NO
=WE(I):: GOTO 1360
1330 IF PA=2 AND WE(I)<>NO T
HEN PAR=PAR+1
1340 IF DR=5 THEN GEW=11 ::
GOTO 1370
1350 IF DR=1 AND PAR=1 THEN
GEW=6 :: GOTO 1370
1360 PA=1 :: NEXT I :: GOTO
1380
1370 IF GEW>0 THEN SUBEXIT
1380 GEW=-((PAR=1)*(1)+(PAR=
2)*(2)+(DR=1)*(5))
1390 SUBEND
1400 SUB FLUSH(FA(),WEG(),GE
W)
```

LISTINGS

```
1410 FOR I=2 TO 5 :: IF FA(1
)=FA(I)THEN 1420 ELSE 1430
1420 NEXT I :: GEW=8 :: FOR
I=1 TO 5 :: WEG(I)=1 :: NEXT
 I
1430 SUBEND
1440 SUB STREET(WE(),WEG(),G
EW)
1450 FOR I=1 TO 5 :: HI(I)=W
E(I):: NEXT I
1460 FOR I=1 TO 5 :: FOR J=I
+1 TO 5
1470 IF HI(I)>HI(J)THEN A=HI
(I):: HI(I)=HI(J):: HI(J)=A
1480 NEXT J :: NEXT I
1490 FOR I=2 TO 5 :: IF HI(1
)<>HI(I)-(I-1)THEN SUBEXIT
1500 NEXT I :: FOR I=1 TO 5
:: WEG(I)=1 :: NEXT I
1510 IF GEW=0 THEN GEW=7 ELS
E IF HI(1)=9 THEN GEW=20 ELS
E GEW=18
1520 SUBEND
1530 SUB HALTEN(WEG())
1540 FOR I=1 TO 5 :: CALL CO
LOR(#I+5,WEG(I)*6+1):: NEXT
I
1550 A=0 :: CALL SOUND(150,4
00,5)
1560 A=A+1 :: IF A>200 THEN
SUBEXIT ELSE CALL KEY(0,K,S)
:: IF S=0 OR K<48 OR K>53 TH
EN 1560
1570 IF K=48 THEN 1600
1580 IF WEG(K-48)=1 THEN WEG
(K-48)=0 ELSE WEG(K-48)=1
1590 CALL COLOR(#(K-43),WEG(
K-48)*6+1):: GOTO 1550
1600 SUBEND
1610 SUB ZWEZIE(WE(),FA(),BL
(,),WEG(),CO())
1620 CALL RUECK(1,5,4,WEG())
1630 CALL RUECK(2,4,6,WEG())
1640 FOR I=1 TO 5
1650 IF WEG(I)=0 THEN CALL Z
UF(A,B,BL(,)):: FA(I)=A :: W
E(I)=B :: BL(A,B)=0
1660 NEXT I
1670 CALL AUSTEI(1,5,4,WE(),
FA(),CO(),WEG()):: CALL AUST
EI(2,4,6,WE(),FA(),CO(),WEG(
))
1680 SUBEND
1690 SUB AUS(GEW,GE(,),ZE)
1700 CALL LOCATE(#11,ZE,GE(G
EW,2)):: IF GEW>10 THEN F=5
ELSE F=7
1710 CALL COLOR(#11,F):: FOR
 I=1 TO 5 :: FOR J=GEW TO GE
W+8 STEP 2 :: CALL LOCATE(#1
1,ZE,GE(J,2)):: CALL KEY(0,K
,S):: IF S<>0 THEN 1730
1720 NEXT J :: NEXT I
1730 CALL PEEK(-31879,A):: B
=-((A<10)+(A<30)+(A<90)+(A<1
70))*2 :: GEW=GEW+B
1740 CALL LOCATE(#11,ZE,GE(G
EW,2)):: SUBEND
1750 SUB RISIKO(GEW,GE(,),CR
E,SER,ZE,S1)
1760 IF GEW>8 THEN F=5 ELSE
F=7
1770 CALL LOCATE(#11,ZE,GE(G
EW,2)):: CALL COLOR(#11,F)::
 FOR I=1 TO 150 :: NEXT I
1780 IF GEW>18 THEN CALL MEL
1 :: K=65 :: GOTO 1810
1790 S2=37 :: S3=GE(GEW+2,2)
:: FOR I=1 TO 35 :: CALL LOC
ATE(#11,ZE,S2):: CALL KEY(0,
K,S):: IF S<>0 THEN 1810
1800 H=S2 :: S2=S3 :: S3=H :
: NEXT I :: K=65
1810 IF K=65 THEN CALL ALL(G
EW,GE(,),CRE,SER,ZE,S1):: SU
BEXIT
1820 IF K=84 THEN CALL TEIL(
GEW,GE(,),CRE,SER,ZE,S1):: G
OTO 1780
1830 CALL PEEK(-31879,A):: I
F A/2=INT(A/2)THEN 1860
1840 CALL LOCATE(#11,ZE,GE(G
EW+2,2)):: GOSUB 400 :: GEW=
GEW+2 :: IF GEW>8 THEN F=5
1850 GOTO 1780
1860 CALL LOCATE(#11,ZE,37):
: GOSUB 410
1870 SUBEND
1880 SUB ALL(GEW,GE(,),CRE,S
ER,ZE,S1)
1890 CALL LOCATE(#11,ZE,GE(G
EW,2)):: CALL MEL :: IF GEW<
9 THEN CRE=CRE+GE(GEW,1)ELSE
 SER=SER+GE(GEW,1):: CRE=CRE
+.3 :: GOSUB 390 :: S1=1
1900 GOSUB 380 :: SUBEND
1910 SUB SERIE(GEW,GE(,),CRE
,SER,S1)
1920 GEW=8 :: IF SER<10 THEN
 CALL RISIKO(GEW,GE(,),CRE,S
ER,173,S1)ELSE CALL COLOR(#1
1,7):: CALL ALL(GEW,GE(,),CR
E,SER,173,S1)
1930 CRE=CRE+.6 :: GOSUB 380
 :: SUBEND
1940 SUB WIN(GEW,GE(,),WEG()
,TE$(),CRE,SER,S1)
1950 IF GEW=0 THEN SUBEXIT
1960 IF GEW/2=INT(GEW/2)THEN
 ZE=173 ELSE ZE=149
1970 FOR I=1 TO 5 :: CALL CO
LOR(#I+5,WEG(I)*2+1):: NEXT
I :: A=LEN(TE$(GEW)):: DISPL
AY AT(18,(28-A)/2)SIZE(A):TE
$(GEW)
1980 IF GEW=1 THEN CALL ZWIL
```

```
L(Z)
1990 IF GEW=11 OR GEW=12 THE
N CALL AUS(GEW,GE(,),ZE):: G
OTO 2040
2000 IF GEW>18 THEN CALL ALL
(GEW,GE(,),CRE,SER,ZE):: SUB
EXIT
2010 IF GEW=16 OR GEW=18 THE
N 2030
2020 IF (SER>0 AND GEW>1)OR(
SER>0 AND Z=4)THEN CALL SERI
E(GEW,GE(,),CRE,SER,S1):: GO
TO 2050 ELSE IF SER>0 THEN S
UBEXIT
2030 IF (Z=4 OR GEW=2)THEN C
ALL AUS(GEW,GE(,),ZE)
2040 IF (Z=4 OR GEW>1)THEN C
ALL RISIKO(GEW,GE(,),CRE,SER
,ZE,S1)
2050 IF Z=4 THEN CALL VCHAR(
19,29,74,4):: Z=0
2060 SUBEND
2070 SUB TEIL(GEW,GE(,),CRE,
SER,ZE,S1)
2080 IF GEW<11 THEN SUBEXIT
2090 GEW=GEW-2 :: CALL LOCAT
E(#11,ZE,GE(GEW,2)):: CALL M
EL :: SER=SER+GE(GEW,1):: IF
 GE(GEW+2,1)/2<>INT(GE(GEW+2
,1)/2)THEN SER=SER+1
2100 GOSUB 390 :: FOR I=1 TO
 150 :: NEXT I :: S1=1 :: SU
BEND
2110 SUB RUECK(A,B,Z,WEG())
2120 FOR I=A TO B STEP 2 ::
IF WEG(I)<>0 THEN 2140
2130 CALL SOUND(250,-4,0)::
CALL COLOR(#I,1):: CALL PATT
ERN(#I,136):: S=-3+(I*6):: F
OR J=0 TO 2 :: CALL VCHAR(Z,
S+J,139,7):: NEXT J :: CALL
COLOR(#I,2)
2140 NEXT I
2150 SUBEND
2160 SUB ZUF(A,B,BL(,))
2170 A=INT(RND*4)+1 :: B=INT
(RND*13)+1 :: IF BL(A,B)=0 T
HEN 2170
2180 SUBEND
2190 SUB ZWILL(A):: A=A+1 ::
 CALL VCHAR(19,29,137,A):: F
OR I=1 TO 200 :: NEXT I :: S
UBEND
2200 SUB MEL :: FOR I=222 TO
 1110 STEP 222 :: CALL SOUND
(150,I,7,I-22,8,I+22,6):: NE
XT I :: SUBEND
2210 SUB MEL1 :: RESTORE 222
0 :: FOR I=1 TO 44 :: READ A
,B :: CALL SOUND(A*125,B,0,B
-2,5,B+2,5):: NEXT I
2220 DATA 2,783,2,783,2,880,
3,739,1,783,2,880,2,987,2,98
7,2,1046,3,987
2230 DATA 1,880,2,783,2,880,
2,783,2,739,4,783,2,43000,2,
1174,2,1174
2240 DATA 2,1174,3,1174,1,10
46,2,987,2,1046,2,1046,2,104
6,3,1046,1,987
2250 DATA 2,880,2,987,1,1046
,1,987,1,880,1,783,3,987,1,1
046,2,1174
2260 DATA 1,1318,1,1174,1,10
46,1,987,2,880,4,783,2,43000
2270 SUBEND
2280 END
```

POKER-BANDIT

Fortsetzung von Seite 40

der Computer hält, können wieder gelöscht werden, nur die entsprechende Zahltaste drücken. Die Eingabe wird mit 0 beendet. Nun startet der Computer die zweite Ziehung. Alle Karten, die nicht gehalten wurden, werden neu ausgeteilt. Die Karten, die nach der zweiten Ziehung einen Gewinn ergeben, werden durch einen grünen Punkt im entsprechenden Zahlfeld angezeigt. Die Art der Gewinnkombination wird geschrieben und der entsprechende Gewinn leuchtet in einer der beiden Risikoleisten auf. Drückt man die Taste "A", so wird der Gewinn ganz angenommen.
Ab 5 bzw. 6 Sonderspielen kann man den Gewinn halbieren. Dazu muß die Taste "T" gedrückt werden. Wird irgendeine beliebige andere Taste gedrückt, so wird der Gewinn riskiert, entweder das Doppelte oder nichts. Das Risikoverhältnis ist 1:1.
In Sonderspielen wird jeder Gewinn auf 3 DM erhöht. Ab 9 Sonderspielen kann wieder riskiert werden.

Zwillinge müssen viermal erscheinen, dann wird ein Gewinn ausgespielt. Auf der rechten Bildschirmseite erscheint für jedes Zwillingspaar ein blaues Feld.

Gewinnplan

Royal Flush = 100 Sonderspiele
Straight Flush = 50 Sonderspiele
Vierling As = 25 Sonderspiele
Vierling = Ausspielung 6–100 Sonderspiele
Drilling As = Ausspielung 5–90 Sonderspiele
Flush = 2.40 DM
Straße = 1.60 DM
Full House = 1.20 DM
Drilling = 0.80 DM
Doppelzwilling = Ausspielung 0.30 DM – 3 Sonderspiele
4mal Zwilling = Ausspielung 0.20 DM – 2 Sonderspiele

Achtung! Wichtig für die Eingabe:
Zeilen 670–750:
Die Punkte sind als 'CTRL K' einzugeben!
Zeilen 840 und 870:
Punkte sind als 'FCTN V' einzugeben!

SURGEON

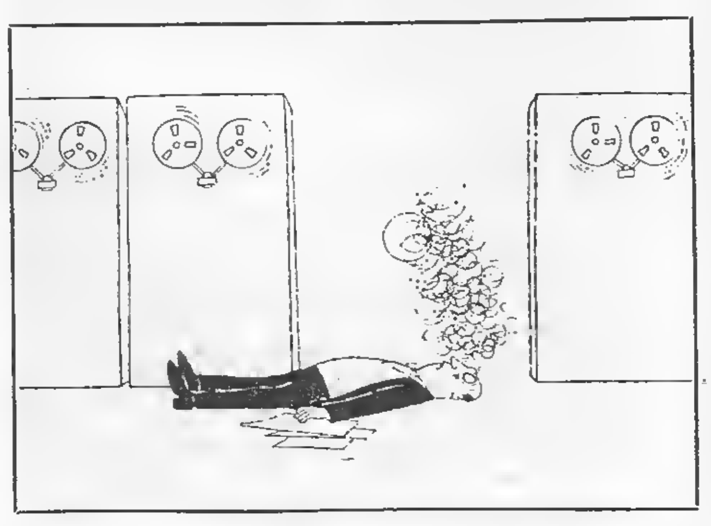
Das vorliegende Programm ist sowohl in Basic als auch in XBasic lauffähig. Sprache ist jedoch nur mit dem TE-II-Modul in Verbindung mit dem Sprachsynthesizer möglich. Falls in Basic oder XBasic auf dem Bildschirm ausgedruckt werden soll, was der Computer mit dem TE-II-Modul + Sprachsynthesizer spricht, sind die folgenden Programmänderungen vorzunehmen:

```
990REM
1040REM
1050FOR I=1 TO LEN
    (E$)
1052IF I>26 THEN
    1060
1054CALL HCHAR
    (23,1+I,ASC(SEGS
    (ES,I,1)))
1056NEXT I
```

Zu Anfang wird gefragt, ob das Spiel mit oder ohne Sprache gespielt werden soll. "J" darf nur mit dem TE-II-Modul+ Sprachsynthesizer gedrückt werden, andernfalls erfolgt Programmabsturz. Danach können Sie wählen zwischen Spielbeschreibung und Spielbeginn. Die Spielbeschreibung ist ausführlich und kann hier daher auf ein Minimum beschränkt werden.

Aufgabe ist es, mit einer Sonde aus der Blutbahn Ihres Patienten verschiedenste Erreger zu entfernen. Dafür steht Ihnen allerdings nur eine beschränkte Zeit zur Verfügung. Der Durchgang durch die Arterien ist schwierig, da sie sich entsprechend dem Herzschlag erweitern und verengen. Dadurch kann die Sonde zerquetscht werden. Nur voll erweiterte Arterien können passiert werden. Um Erreger aufzunehmen, ist die Sonde über diese hinwegzuführen. Daraufhin erscheint der Hinweis, wo die Sonde aus der Blutbahn herauszuführen ist, um den Erreger abzulegen. Je nach Leistung (Zeit, Zahl der entfernten Erreger) wird Ihnen ein Titel verliehen. Die folgenden Grade sind möglich: Anfänger, Student, Intern, Surgeon. Die Steuerung erfolgt mit Joystick 1. Um die Sonde in alle acht Richtungen bewegen zu können, vergessen Sie nicht, die Alpha-Lock-Taste auszurasten!

Neben dem Blutgefäßsystem und dem Desinfektionseimer werden auf dem Bildschirm unter dem TI-Symbol die entfernten Fremdkörper, unten in der Mitte die aktuelle Punktzahl, rechts daneben der HI-Score angezeigt.



```
10 REM********************
11 REM*      SURGEON      *
12 REM*                   *
13 REM*  Copyright  by    *
14 REM*                   *
15 REM* Manfred Lipowski *
16 REM*                   *
17 REM*Benoetigte Geraete*
18 REM* TI99/4A  Konsole *
19 REM*   Joystick 1      *
20 REM*   optional:       *
21 REM* TE-II-Modul plus *
22 REM*Sprach-Synthesizer*
23 REM*                   *
24 REM* Speicherbelegung *
25 REM*    11033 Bytes    *
26 REM********************
27 REM
100 CALL CLEAR
110 RANDOMIZE
120 SPR=0
130 CALL SCREEN(11)
140 FOR A=1 TO 12
150 CALL COLOR(A,16,7)
160 NEXT A
170 PRINT : : : : : :
180 PRINT "........S U R G E
 O N": :"........(TE-2  MODU
L)": :".....MIT SPRACHE ? (J
/N)": : : : : : : : : : : :
: :"....(c) M.LIPOWSKI 1985"
190 CALL KEY(0,K,S)
200 IF S=0 THEN 190
210 IF K=78 THEN 260
220 IF K=74 THEN 230 ELSE 19
0
230 OPEN #1:"SPEECH",OUTPUT
240 PRINT #1:"^WELLCOME TO S
URGEON . . PLEASE ^WAIT"
250 SPR=1
260 CALL CLEAR
270 CALL SCREEN(1)
280 CALL COLOR(1,1,1)
290 DATA 33,00000000031F7FFF
,34,00000000FFFFFFFF,35,0000
0000C0F8FEFF,40,FF7F1F03,41,
FFFEF8C,42,FFFFFFFF
300 DATA 43,0,104,0039203030
2039,105,007C101010101,106,0
012A04242A212,112,1C161210F1
81C06
310 DATA 113,203C0603010101,
114,C0C018CCE6E2C2D2,115,F6E
44C183060C,116,0004066585846
4,117,0048C84848484E
320 DATA 136,0000183C3C18,13
7,081C7E3C183C381,138,00040C
18306,139,000014183C3,140,00
245A66182418
330 DATA 141,001C3464407018,
142,0000000008,143,000030281
40C,120,FFFFFFFFFFFFFFFF
```

LISTINGS

```
340 DATA 96,0008081C081C36,9
7,000002167C1602,98,0040683E
684,99,00361C081C0808,123,FF
FFFFFFFFFFFFFF
350 DATA 88,F8E0C0808,89,000
0008080C0E0F8,90,1F07030101,
91,0000000101030071F,92,F8E0C
08080C0E0F8
360 DATA 93,1F0703010103071F
,95,00FF,37,0044442810284444
,38,00444428101010 1,39,007C0
4081020407C
370 RESTORE 290
380 FOR A=1 TO 40
390 READ B,B$
400 CALL CHAR(B,B$)
410 NEXT A
420 D$(1)="C3E7FFFFFFFFE7C3"
430 D$(2)="8181C3FFFFC38181"
440 D$(3)="000081FFFF810000"
450 D$(4)="000000FFFF000000"
460 D$(5)="000081FFFF810000"
470 D$(6)="8181C3FFFFC38181"
480 D$(7)="C3E7FFFFFFFFE7C3"
490 F$(1)="FFFF7E3C3C7EFFFF"
500 F$(2)="FF3C181818183CFF"
510 F$(3)="3C1818181818183C"
520 F$(4)="1818181818181818"
530 F$(5)="3C1818181818183C"
540 F$(6)="FF3C181818183CFF"
550 F$(7)="FFFF7E3C3C7EFFFF"
560 CALL CHAR(121,D$(4))
570 CALL CHAR(122,F$(4))
580 CALL SCREEN(12)
590 DATA 1,6,12,2,6,14,3,5,1
,4,5,1,5,2,1,6,2,1,7,2,1,8,7
,1,9,16,7,10,1,1
600 DATA 11,13,1,12,7,1,14,1
5,7
610 CALL SOUND(99,-1,1)
620 FOR A=1 TO 9
630 CALL COLOR(A,2,1)
640 NEXT A
650 PRINT "SPIELANLEITUNG ?
J N": : : : : : : : : : : :
: :
660 CALL KEY(0,K,S)
670 IF S=0 THEN 660
680 IF K=74 THEN 690 ELSE 70
0
690 GOSUB 4160
700 RESTORE 590
710 FOR A=1 TO 13
720 READ B,C,D
730 CALL COLOR(B,C,D)
740 NEXT A
750 CALL HCHAR(24,22,72)
760 CALL HCHAR(24,23,83)
770 CALL HCHAR(24,24,58)
780 CALL HCHAR(24,30,43,3)
790 CALL HCHAR(23,30,43,3)
800 DATA "tu.........SURGEON
..........pr","..........__
_____...........qs","xxxxxxxx
xyxxxxxyxxxxxxxxyxxx"
810 DATA "xXZxXZx\...]xX....
.ZxX...Zx","x..xY[xxxyxxx...
....xY....x","z..xxxxX...Zx.
.......xxx...z"
820 DATA "xY....z.....z.....
..ZxX...x","xxxxxxxxxxY[xY..
....[x...[x","........Zxxxxx
xxxxyxxx xxxx"
830 DATA "xxxxyxxxY....ZxX..
......xX","x\....Zxxx...x...
......x..hji","xxyxxx...ZxY [
xY......[xY"
840 DATA "xX xxx...[xxyxxxyx
yxyxxxxxxxx{","xY....[xxxxX
ZxX......ZxX","xxxxyxxxX....
[xY.......xY"
850 DATA "........[xxxxxxxxx
yxxx xxxx","xxxxxxxxxxXZxX..
....Zx...Zx","xX....z.....x.
.......[xY...x"
860 DATA "z..xxxxY....z.....
..xxx...z","x..xXZxxxyxxx...
....xX....x","xY[xY[x\...]xY
.....[xY...[x ! #"
870 DATA "xxxxxxxxxyxxxxxyxx
xxxxxyxxx (*)"
880 RESTORE 800
890 FOR A=1 TO 22
900 READ A$
910 FOR B=1 TO LEN(A$)
920 CALL HCHAR(A,1+B,ASC(SEG
$(A$,B,1)))
930 NEXT B
940 NEXT A
950 CALL HCHAR(21,31,34)
960 E$="^WELLCOME TO SURGEON
 . ^>DOCTOR"
970 GOSUB 990
980 GOTO 1070
990 IF SPR=0 THEN 1000 ELSE
1050
1000 TON=INT(RND*400)+220
1010 FOR T=0 TO 30 STEP 2.5
1020 CALL SOUND(-99,TON,T)
1030 NEXT T
1040 RETURN
1050 PRINT #1:E$
1060 RETURN
1070 Z=0
1080 ML1=0
1090 SC=0
1100 ZZ=120
1110 BA=0
1120 LP=0
1130 CALL GCHAR(1,2,S)
1140 IF S<>116 THEN 100
1150 CALL HCHAR(23,1,32,28)
1160 CALL HCHAR(4,29,32,4)
1170 CALL HCHAR(24,1,32,19)
1180 CALL COLOR(9,16,7)
1190 CALL COLOR(10,1,1)
```

LISTINGS

```
1200 CALL HCHAR(13,31,123)
1210 TE2=0
1220 XX=13
1230 YY=30
1240 CALL HCHAR(XX,YY,97)
1250 E$="YOUR ^>ROBOT PROBE"
1260 GOSUB 990
1270 GOSUB 3330
1280 Z=Z+1
1290 TE2=TE2+1
1300 CALL CHAR(121,D$(Z))
1310 CALL CHAR(122,F$(Z))
1320 IF Z<7 THEN 1340
1330 Z=0
1340 IF BA=0 THEN 1360
1350 IF (Z=2)+(Z=3)+(Z=4)+(Z
=5)+(Z=6)THEN 1860
1360 CALL SOUND(-4250,-5,Z*4
)
1370 CALL JOYST(1,X,Y)
1380 IF TE2>150 THEN 2520
1390 IF (X=0)*(Y=0)THEN 1280
1400 IF X=-4 THEN 1450
1410 IF X=4 THEN 1470
1420 IF Y=-4 THEN 1490
1430 IF Y=4 THEN 1510
1440 GOTO 1370
1450 P=97
1460 GOTO 1520
1470 P=98
1480 GOTO 1520
1490 P=99
1500 GOTO 1520
1510 P=96
1520 CALL HCHAR(XX,YY,ZZ)
1530 XX=XX+(Y/-4)
1540 YY=YY+(X/4)
1550 CALL GCHAR(XX,YY,ZZ)
1560 IF ZZ=120 THEN 1570 ELS
E 1590
1570 BA=0
1580 GOTO 2500
1590 IF (ZZ=121)+(ZZ=122)THE
N 1600 ELSE 1630
1600 IF (Z=1)+(Z=7)+(Z=0)THE
N 1610 ELSE 1860
1610 BA=1
1620 GOTO 2500
1630 IF ZZ>135 THEN 2560
1640 IF (TI=1)*(ZZ=123)THEN
2660
1650 CALL HCHAR(XX,YY,P)
1660 FOR A=0 TO 30 STEP 5
1670 CALL HCHAR(XX,YY,96)
1680 GOSUB 3980
1690 CALL COLOR(9,14,1)
1700 CALL HCHAR(XX,YY,98)
1710 GOSUB 3980
1720 CALL SOUND(-999,-7,A)
1730 CALL HCHAR(XX,YY,99)
1740 GOSUB 3980
1750 CALL COLOR(9,16,1)
1760 CALL HCHAR(XX,YY,97)
1770 GOSUB 3980
1780 NEXT A
1790 E$="^>AUTCH"
1800 GOSUB 990
1810 IF ZZ<88 THEN 1820 ELSE
 1830
1820 ZZ=32
1830 CALL HCHAR(XX,YY,ZZ)
1840 CALL HCHAR(L,M,120)
1850 GOTO 2040
1860 CALL HCHAR(XX,YY,P)
1870 FOR A=0 TO 30 STEP 5
1880 CALL HCHAR(XX,YY,96)
1890 GOSUB 3980
1900 CALL COLOR(9,11,7)
1910 CALL HCHAR(XX,YY,97)
1920 GOSUB 3980
1930 CALL SOUND(-999,-6,A)
1940 CALL HCHAR(XX,YY,99)
1950 GOSUB 3980
1960 CALL COLOR(9,16,7)
1970 CALL HCHAR(XX,YY,98)
1980 GOSUB 3980
1990 NEXT A
2000 E$="^>_OUPS"
2010 GOSUB 990
2020 CALL HCHAR(XX,YY,ZZ)
2030 CALL HCHAR(L,M,120)
2040 G=INT(RND*3)+1
2050 ON G GOTO 2060,2090,212
0
2060 E$="WHAT HAVE ^YOU DONE
?"
2070 GOSUB 990
2080 GOTO 2140
2090 E$="PLEASE ^ HELP ME DO
CTOR"
2100 GOSUB 990
2110 GOTO 2140
2120 E$="^DOC . PLEASE ^>CAU
TION"
2130 GOSUB 990
2140 E$="NEUES SPIEL ? J N"
2150 TY=23
2160 GOSUB 4010
2170 IF ML1>6 THEN 2240
2180 E$="SORRY . PLAY AGAIN
. HEAD ^HIGH"
2190 GOSUB 990
2200 E$="ANFAENGER"
2210 TY=24
2220 GOSUB 4010
2230 GOTO 2430
2240 IF ML1>13 THEN 2310
2250 E$="YOU ARE A . ^>STUDE
NT"
2260 GOSUB 990
2270 E$="STUDENT"
2280 TY=24
2290 GOSUB 4010
2300 GOTO 2430
2310 IF ML1>19 THEN 2380
2320 E$="YOU ARE A ^>INTERN"
```

LISTINGS

```
2330 GOSUB 990
2340 E$="INTERN"
2350 TY=24
2360 GOSUB 4010
2370 GOTO 2430
2380 E$="SURGEON"
2390 IF ML2<>1 THEN 2340
2400 TY=24
2410 GOSUB 4010
2420 GOSUB 4060
2430 E$="^NEW ^>GAME . J OR
^N ?"
2440 GOSUB 990
2450 CALL KEY(0,K,S)
2460 IF S=0 THEN 2450
2470 IF (K=74)+(K=106)THEN 1
070
2480 CALL CLEAR
2490 STOP
2500 CALL HCHAR(XX,YY,P)
2510 GOTO 1280
2520 CALL SOUND(-1,-1,30)
2530 E$="^TIME IS ^>OUT ^>DO
CTOR"
2540 GOSUB 990
2550 GOTO 1860
2560 ZZ=120
2570 CALL HCHAR(XX,YY,P)
2580 TI=1
2590 CALL SOUND(-1,-1,30)
2600 CALL COLOR(10,16,6)
2610 IF SPR=0 THEN 2640
2620 PRINT #1:"^>OUPS ^DOCTO
R"
2630 GOTO 1280
2640 CALL SOUND(-99,110,9,11
2,7)
2650 GOTO 1280
2660 IF SPR=0 THEN 2820
2670 CALL SOUND(-1,-1,30)
2680 GOSUB 3890
2690 CALL HCHAR(XX,YY,P)
2700 QT=INT(RND*5)+1
2710 ON QT GOTO 2720,2740,27
60,2780,2800
2720 PRINT #1:"EXIT . DOCTER
"
2730 GOTO 2810
2740 PRINT #1:"RIGHT DOC"
2750 GOTO 2810
2760 PRINT #1:"YOU ARE VERY
GOOD DOCTOR"
2770 GOTO 2810
2780 PRINT #1:"FINE . YOU HE
LP ME DOCTOR"
2790 GOTO 2810
2800 PRINT #1:"YOU ARE THE B
EST DOC"
2810 GOSUB 3920
2820 CALL COLOR(14,2,1)
2830 CALL HCHAR(XX,YY,P)
2840 FOR LM=1 TO 7
2850 CALL SOUND(-299,800-LM*
10,LM)
2860 CALL HCHAR(13+LM,31,ML)
2870 CALL HCHAR(13+LM,31,32)
2880 NEXT LM
2890 FOR LM=0 TO 30 STEP 2.5
2900 CALL SOUND(-99,-7,LM)
2910 NEXT LM
2920 GOSUB 3890
2930 GOSUB 990
2940 E$="IN ^>THE . ^DUSTBIN
"
2950 GOSUB 990
2960 GOSUB 3920
2970 TE3=150
2980 TE3=TE3-TE2
2990 SC=SC+LP+TE3
3000 FOR I=1 TO LEN(STR$(SC)
)
3010 CALL HCHAR(24,14+I,ASC(
SEG$(STR$(SC),I,1)))
3020 IF SC<HS THEN 3050
3030 HS=SC
3040 CALL HCHAR(24,24+I,ASC(
SEG$(STR$(SC),I,1)))
3050 CALL SOUND(-1,-1,9)
3060 NEXT I
3070 ML1=ML1+1
3080 FOR I=1 TO LEN(STR$(ML1
))
3090 CALL HCHAR(4,29+I,ASC(S
EG$(STR$(ML1),I,1)))
3100 NEXT I
3110 IF ML1<=19 THEN 3180
3120 GOSUB 3890
3130 IF SPR=0 THEN 3150
3140 PRINT #1:"THIS GAME IS
OVER . . YOU ARE THE SURGEON
"
3150 GOSUB 3920
3160 ML2=1
3170 GOTO 2140
3180 I=INT(RND*3)+1
3190 GOSUB 3950
3200 ON I GOTO 3210,3240,327
0
3210 E$="^>FINE DOC"
3220 GOSUB 990
3230 GOTO 3290
3240 E$="^THANKS DOC"
3250 GOSUB 990
3260 GOTO 3290
3270 E$="^THANKS ^>DOCTOR"
3280 GOSUB 990
3290 GOSUB 3920
3300 E$="O+K LETS+^GO . ^NEX
T ^>ROUND ^>DOCTOR"
3310 GOSUB 990
3320 GOTO 1180
3330 M=INT(RND*20)+1
3340 L=INT(RND*20)+2
3350 CALL GCHAR(L,M,U)
3360 IF U<>120 THEN 3330
3370 CALL GCHAR(L,M+1,U)
```

LISTINGS

```
3380 CALL GCHAR(L,M-1,U1)
3390 CALL GCHAR(L+1,M,U2)
3400 CALL GCHAR(L-1,M,U3)
3410 IF (U=121)+(U=122)+(U1=
121)+(U1=122)+(U2=121)+(U2=1
22)+(U3=121)+(U3=122)THEN 33
30
3420 ML=INT(RND*8)+1
3430 ON ML GOTO 3440,3490,35
40,3590,3640,3690,3740,3790
3440 ML=136
3450 LP=100
3460 FA=16
3470 E$="^VIRUS"
3480 GOTO 3830
3490 ML=137
3500 LP=150
3510 FA=16
3520 E$="WHITE ^VIRUS"
3530 GOTO 3830
3540 ML=138
3550 LP=50
3560 FA=12
3570 E$="^>WORM"
3580 GOTO 3830
3590 ML=139
3600 LP=200
3610 FA=15
3620 E$="^LITTLE TUMOR"
3630 GOTO 3830
3640 ML=140
3650 LP=400
3660 FA=10
3670 E$="^GREAT ^TUMOR"
3680 GOTO 3830
3690 ML=141
3700 LP=300
3710 FA=16
3720 E$="^TAPEWORM"
3730 GOTO 3830
3740 ML=142
3750 LP=50
3760 FA=4
3770 E$="^>BACTERIA"
3780 GOTO 3830
3790 ML=143
3800 LP=350
3810 FA=12
3820 E$="^GREAT ^>BACTERIA"
3830 CALL HCHAR(L,M,ML)
3846 CALL COLOR(14,FA,7)
3850 GOSUB 3890
3860 GOSUB 990
3870 GOSUB 3920
3880 RETURN
3890 IF SPR=0 THEN 3910
3900 PRINT #1:"//63 255"
3910 RETURN
3920 IF SPR=0 THEN 3940
3930 PRINT #1:"//40 64"
3940 RETURN
3950 IF SPR=0 THEN 3970
3960 PRINT #1:"//1 0.16"
3970 RETURN
3980 FOR ZE=1 TO 16
3990 NEXT ZE
4000 RETURN
4010 FOR I=1 TO LEN(E$)
4020 CALL HCHAR(TY,2+I,ASC(S
EG$(E$,I,1)))
4030 CALL SOUND(1,-3,19)
4040 NEXT I
4050 RETURN
4060 DATA 6,262,2,196,6,262,
2,196,2,262,2,196,2,262,2,33
0,6,392
4070 DATA 6,349,2,294,6,349,
2,294,2,349,2,294,2,247,2,29
4,8,196
4080 IF ML1<19 THEN 4150
4090 RESTORE 4060
4100 FOR D=1 TO 18
4110 READ A,B
4120 CALL SOUND(A*99,B,0,B+B
,1)
4130 NEXT D
4140 CALL SOUND(1,-1,30)
4150 RETURN
4160 PRINT ".........S U R G
E O N":"........___________
_"
4170 PRINT "ERREICHEN SIE DE
N TITEL": :"EINES SURGEON.DI
ES IST": :"JEDOCH KEINE LEIC
HTE AUFGABE"
4180 PRINT :"VERSUCHEN SIE V
IEREN,TUMORE": :"WUERMER UND
 BACTERIEN AUS": :"DEM ARTER
IENS&STEM IHRES"
4190 PRINT :"PATIENTEN MIT D
ER ROBOTSONDE": :"'U ENTFERN
EN. ...........>>": : : :
4200 CALL KEY(0,K,S)
4210 IF S=0 THEN 4200
4220 CALL CLEAR
4230 PRINT "HABEN SIE NUN ET
WAS ENTFERNT": :"LEUCHTET DA
S E%IT SCHILD AUF": :"BRINGE
N SIE DEN FREMDKOERPER"
4240 PRINT :"BIS 'UM ENDE DE
R BLUTBAHN.": :"DORT WIRD ER
 DANN AUTOMA": :"TISCH IN DE
N ABFALLEIMER"
4250 PRINT :"GEWORFEN.SIE BE
KOMMEN DANN": :"DIE JEWEILIG
E PUNKT'AHL DES": :"FREMDKOE
RPERS PLUS DIE VER"
4260 PRINT :"BLEIBENDE 'EIT
GUTGESCHRIE": :"BEN. .......
.........>>"
4270 CALL KEY(0,K,S)
4280 IF S=0 THEN 4270
4290 CALL CLEAR
4300 PRINT "VERLASSEN SIE DI
E BLUTBAHN": :"IST DAS SPIEL
 'UENDE UND IHR": :"ERREICHT
```

```
ER SPIELGRAD WIRD"
4310 PRINT :"ANGE'EIGT.ANFAE
NGER,STUDENT": :"INTERN ODER
 SURGEON.": :"IHRE ROBOTSOND
E IST IN JEDE".
4320 PRINT :"RICHTUNG STEUER
BAR.ALSO": :"VORSICHT... ...
...........>>"
4330 CALL KEY(0,K,S)
4340 IF S=0 THEN 4330
4350 CALL CLEAR
4360 PRINT "";CHR$(136);" VI
RUS.............100 P": :"";
CHR$(137);" WEISSER VIRUS...
..150 P"
4370 PRINT :"";CHR$(138);" W
ORM...............50 P": :""
;CHR$(139);" KLEINER TUMOR..
...200 P"
4380 PRINT :"";CHR$(140);" G
ROSSER TUMOR.....400 P": :""
;CHR$(141);" TAPEWORM.......
...300 P"
4390 PRINT :"";CHR$(142);" B
ACTERIE...........50 P": :""
;CHR$(143);" GROSSE BACTERIE
...350 P"
4400 CALL KEY(0,K,S)
4410 IF S=0 THEN 4400
4420 CALL CLEAR
4430 RETURN
```

LISTINGS

BÖRSE

CLUB- ECKE

Beginnen möchten wir heute mit einer Mitteilung des Computer Clubs Karben, dessen Einleitung nichts hinzuzufügen ist:
Da die Clubecke in der letzten TI-REVUE-Ausgabe wirklich sehr klein war, will sich der Computer-Club Karben mal wieder an dieser Stelle melden.
Leider beschränken sich unsere Clubaktivitäten in der Hauptsache auf unsere Club-Zeitung, den TI-REPORT, und dem Programmtausch. Doch wir hoffen, durch weiteren Mitgliederzuwachs auch von mehr Clubaktivitäten berichten zu können.
Die Clubzeitung erscheint alle drei Monate. Layout und alles das, was für die Entstehung der Zeitung notwendig ist, wird vom TI erledigt.
Im TI-REPORT kann man Testberichte, Programmierhilfen, Preislisten usw. finden. Auf den 26 „Drucker-Seiten" der ersten Ausgabe dieses Jahres ist sicher für jeden User etwas dabei. Besonders interessant ist der CCK für den, der auch aufs Geld achten muß (und das gilt ja wohl für jeden!), denn immer mehr Firmen bieten für Clubmitglieder Rabatte an.
Auch ein neues Adventure für das Adventuremodul wurde von den Mitgliedern des Clubs entwickelt, "Last Mission" ist für DM 30,– zu haben. Für Clubmitglieder kostet es DM 20,–.
Der Beitrag im CCK beträgt für 3 Monate DM 4,–; billiger ist es kaum zu haben. Darin enthalten ist die TI-REPORT-Ausgabe für diesen Zeitraum und die Programmtauschliste.
Der Programmtausch (ganz legal!) kostet – außer Porto – nichts.
Weitere Informationen über den Club gibt es für 80 Pf. Rückporto.
Unsere Adresse:
Computer-Club Karben
Jochen Schmitt
Am Hellenberg 24
6367 Karben 1

Die nächste Nachricht kommt aus dem Norden, wo in der Gegend um Pinneberg TI-User zur Gründung eines Clubs gesucht werden:

Im Kreis Pinneberg (Norddeutschland) soll ein neuer Computerclub entstehen. Wir wollen uns hauptsächlich mit dem TI beschäftigen, aber auch Interessenten mit anderen Systemen können sich gerne bei uns melden (ein SX-64 ist vorhanden).
Unser Hauptziel ist es, eine Clubzeitschrift zu entwerfen und einen verstärkten Software-Austausch einzuleiten. Ferner denken wir an ein regelmäßiges Treffen und evtl. an einen Datenaustausch mit Hilfe eines Akkustikkopplers. Wir sind bemüht, einen guten Kontakt zwischen TI-Usern herzustellen, der auch nach längerer Zeit nicht vergessen wird.
Wer sich bei uns melden möchte, sollte mindestens eine TI-Konsole, einen Kassettenrekorder und das Extended-Basic Modul besitzen.
Da ich selbst eine komplette Anlage habe, d.h. mit Diskettenstation, 32-KB, Editor/Assembler usw., sind auch TI-Fans mit anders ausgestatteten Anlagen herzlich willkommen. Sehr interessiert sind wir auch an Usern mit neuem Zubehör, wie z.B. von Atronic oder Radix (CPS99, 256 KB usw.).
Wir sind bis jetzt leider nur drei aktive TI-Anwender, suchen aber Kontakte in der gesamten Bundesrepublik sowie in allen anderen Ländern.
Ob alle diese Ziele zu verwirklichen sind, wissen wir erst, wenn bekannt ist, wieviele Interessenten es gibt.
Wer sich also bei uns melden möchte, schreibt bitte an:
T.I.S.S.
Thorsten Ziehm
An der Au 17
2081 Haseldorf



Die letzte Mitteilung für heute kommt vom Club der Toten:

Der Club der Toten (CDT) zieht Zwischenbilanz.
Immer mehr Clubs beschäftigen sich entweder mit anderen Computern oder hören ganz auf. Schade!
Aber der TI-User muß nicht verzweifeln, denn es gibt da ja noch den Club der Toten und wir bieten weiterhin: Clubinfo, Softwaretausch, Einkaufsgemeinschaften, nach Wunsch Treffen, usw. (und das für 12,– DM/Jahr). Aber jetzt kommt der Hammer:
Der Club der Toten bietet ein starkes Computermagazin für TI-99/4A und VIC=20!!! Es heißt 2099er!!!
Wie der Name entstanden ist, kann man sich ja wohl denken. (20 von VIC=20 und 99 von TI-99/4A und er von 64er)
Wir wollen aber nicht etwa der TI-REVUE Konkurrenz machen. sondern noch mehr Literatur für den TI-User bieten, der es wirklich nötig hat.
Die erste Ausgabe, einigen von Euch wohl schon bekannt, kam vor 2 Monaten heraus, jetzt gibt es schon Ausgabe Nr. 2.
Wir wollen aber nicht allen Nicht-CDT-Mitgliedern die 2099er vorenthalten, daher machen wir folgendes Angebot:
Wer uns zwei 50 Pfg-Marken schickt (Porto und Verpackung) erhält die 2099er Ausgabe Nr. 2.
Unsere Adresse:
2099er
Am Hang 5
2054 Geesthacht
In der 2099er sind auch Angaben zum Club der Toten gemacht.

Übrigens solltet Ihr natürlich allen Anfragen an die Clubs Rückporto beilegen. Das war's dann für heute, bis zum nächsten Mal auf der Clubseite. Noch ein kurzer Brief für alle Clubs zum Nachdenken:

Wie ich der TI-REVUE entnehme, gründen und treffen sich immer wieder neue TI-99-Clubs, schön für alle! Der TI ist und bleibt ein guter und leistungsfähiger Computer und man kennt inzwischen ja schon Leute, die derart auf ihn schwören, daß sie mit einem Apple oder anderem PC nicht tauschen würden.
Es ist auch schön, wenn Erfahrungen mit dem TI ausgetauscht werden; was mich nur so ärgert ist, daß jeder Club sein eigenes Süppchen kocht, sprich: eine eigene Clubzeitung macht. Dadurch, so meine ich, ist uns allen nicht gedient, schon gar nicht Leuten, die weit vom nächsten Club entfernt wohnen und sich alles selbst erarbeiten müssen.
Meine Bitte: Veröffentlicht Euer Wissen und Eure Erfahrungen, Tips und Tricks doch in einer käuflichen Zeitung, wie der TI-REVUE.
Gerold Saß, Berlin

TI-FORTH: EINE MÄCHTIGE PROGRAMMIER-SPRACHE!

Dieser Artikel, und die eventuell noch folgenden, versteht sich als eine Art Einführung in TI-Forth. Es wird jedoch vorausgesetzt, daß man schon einige Erfahrungen in der Arbeit mit TI-Forth hat. Anhand einiger kleiner, aber nützlichen Routinen, und später mit größeren Programmen, werden verschiedene Aspekte des TI-Forth besprochen.

Das Wichtigste in Forth und auch eine der größten Unterschiede zu Basic, Pascal und anderen Programmiersprachen ist wohl der Stapel (= Stack) und die Stapelorientierung.

Es gibt drei solcher Stapel, den Datenstapel, den Returnstapel und den Wörterstapel (= Dictionary). Alle Rechenoperationen verlaufen über den Datenstapel. Der Returnstapel enthält die Rückkehradressen und die Zwischenwerte, und der Wörterstapel enthält die Forthwörter, welche die Routinen und Programme sind.
Es gibt verschiedene Wörter, die den Datenstapel manipulieren, aber drei der meistgebrauchten Forthwörter sind in TI-Forth nicht enthalten. Es handelt sich dabei um die Befehle DEPTH, PICK und ROLL. Wir werden diese angesprochenen Wörter selbst definieren. Diese Definitionen finden wir auf Screen 1.

DEPTH (———n) legt auf den Datenstapel die Anzahl der Elemente, sprich Zahlen des Datenstapels, vor dem Anruf von DEPTH. Wollen wir also wissen, wie groß unser Datenstapel im Moment ist, dann tippen wir: [DEPTH . <CR>]. Alles was zwischen den eckigen Klammern steht, muß eingegeben werden. Der Punkt bedeutet für TI-Forth, daß es den obersten Stapelwert auf dem Bildschirm ausgeben soll. Nachdem der Befehl mit ENTER bestätigt worden ist, antwortet TI-Forth zum Beispiel mit der Meldung: 23 OK. Das heißt, daß es 23 Zahlen (eine Zahl entspricht 2 Bytes) auf den Datenstapel vor den Anruf von DEPTH waren. Da wir die Zahl, die DEPTH auf den Datenstapel legt, wieder von ihm genommen haben (mit .), sind es immer noch 23. Wenn man wissen will, welche Zahlen auf den Datenstapel stehen, dann kann man dies mit [.S <CR >] erreichen. Dieses Wort steht auf dem Screen mit dem Namen -DUMP.

DEPTH arbeitet wie folgt: Zuerst wird angegeben, daß man ein neues Wort definieren will, welches mit : geschieht. Dann folgt der Name des neuen Befehls, das ist in diesem Fall DEPTH.
Danach erst wird die unterste Stapelposition und dann die oberste Stapelposition geholt, dabei lauten die Befehle S0 und SP. Diese werden subtrahiert und durch 2 dividiert, denn jede Zahl besteht aus 2 Bytes. Das Ergebnis bleibt oben auf dem Datenstapel liegen.
Dann muß nur noch das neue Wort abgeschlossen werden, was mit einem Semicolon (;) erreicht wird.
PICK (n1———n2) kopiert die n1. Zahl des Datenstapels nach oben. PICK arbeitet wie folgt: Nach Angabe des Namens des neuen Wortes wird die Zahl n1 mit zwei multipliziert mit DUP + (addieren ist schneller als multiplizieren). Das Ergebnis wird zum Stackpointer addiert und von dieser Adresse werden zwei Bytes geholt mit dem Wort .

Die Programmiersprache FORTH ist in vielerlei Hinsicht revolutionär. Folgende Vorzüge von FORTH sind dabei besonders hervorzuheben:

- FORTH ist schnell. Im Schnitt laufen FORTH-Programme mehr als 10 mal so schnell wie entsprechende BASIC-Programme.
- FORTH erfordert sehr wenig Speicherplatz. Die meisten Compiler blähen die Programmlänge enorm auf. Aufgrund der manchmal unglaublichen Codeeffizienz von FORTH lassen sich sehr leistungsfähige Programme auf relativ wenig Speicherplatz unterbringen. (In einer Projektstudie wurde festgestellt, daß ein kompiliertes FORTRAN-Programm von 64K in FORTH nur etwa 16K benötigte).
- FORTH ist strukturiert. Wie alle moderneren Hochsprachen (PASCAL, C) enthält FORTH schachtelbare Strukturen für Verzweigungen und Schleifenbildung.
- FORTH ist sowohl eine Interpreter- als auch eine Compilersprache. Die Programme können genauso einfach interaktiv entwickelt werden wie mit einem BASIC-Interpreter. Der genial einfache FORTH-Compiler sorgt dennoch für eine wesentlich höhere Ablaufgeschwindigkeit bei maximaler Codeeffizienz.
- FORTH ist maschinennah. Daher kann FORTH in vielen Fällen die Assemblerprogrammierung vollständig ersetzen.
- FORTH kann in beliebiger Richtung erweitert werden und so als problemorientierte Sprache für das jeweilige Anwendungsfeld optimiert werden.
- FORTH ist eine offene Sprache. Da FORTH dem Programmierer keine Zwangsjacken anlegt, können nahezu beliebige Daten- und Programmstrukturen verwirklicht werden.

SERVICE

ROLL (n———) rotiert das n. Wort des Datenstapels nach oben. ROLL arbeitet wie folgt: Erst wird die n. Zahl nach oben kopiert, aber damit man die Zahl n nicht verliert, wird diese erst mit DUP dupliziert. Das hat aber zur Folge, daß eine Zahl mehr auf dem Stapel steht, daher fängt man an mit 1+. Dann geht's weiter wie bei PICK mit DUP + SP + DUP, das letzte DUP dient dazu, daß man die Adresse der n. Zahl nicht verliert. Jetzt rotiert man die obersten drei Zahlen mit ROT ROT, so daß die Adresse der n. Zahl oben steht. Diese wird mit 2 subtrahiert und damit zeigt unsere Adresse nach die n-1. Zahl. Mit SWAP holt man jetzt wieder 2n+1 nach oben und geht damit in eine Schleife (n mal, von 0 bis 2n-1). In der Schleife wird von der n-1. Zahl an jede Zahl um 2 Bytes nach unten in den Datenstapel verschoben mit DUP DUP SWAP 2+ !. Die Schleife wird beendet mit 2 +LOOP, das heißt, die Indexvariable wird um 2 erhöht. Wenn man aus der Schleife kommt, wird mit DROP DROP die übriggebliebene Adresse gelöscht und die übriggebliebene Zahl, weil man alle Zahlen nach unten verschoben hat. Beide Wörter, PICK und ROLL, haben den gleichen Effekt, mit dem Unterschied, daß PICK die Zahl in dem Datenstapel stehen läßt, während ROLL sie herausholt.
Man kann beide Wörter noch mit Fehlermeldungen ausstatten, wenn sie mittels falscher Parameter (Screen 2) aufgerufen worden sind. Man muß dann aber im Forthstil die Fehlermeldung auf Screen 4 oder 5 eintragen (Screen 4 und 5). Mit den Fehlermeldungen auf Screen 4 und 5 sind wir bei zwei nützlichen Routinen angelangt (Screen 3). Die Meldungen dafür stehen in Screen 4, Zeile 13 und 14.

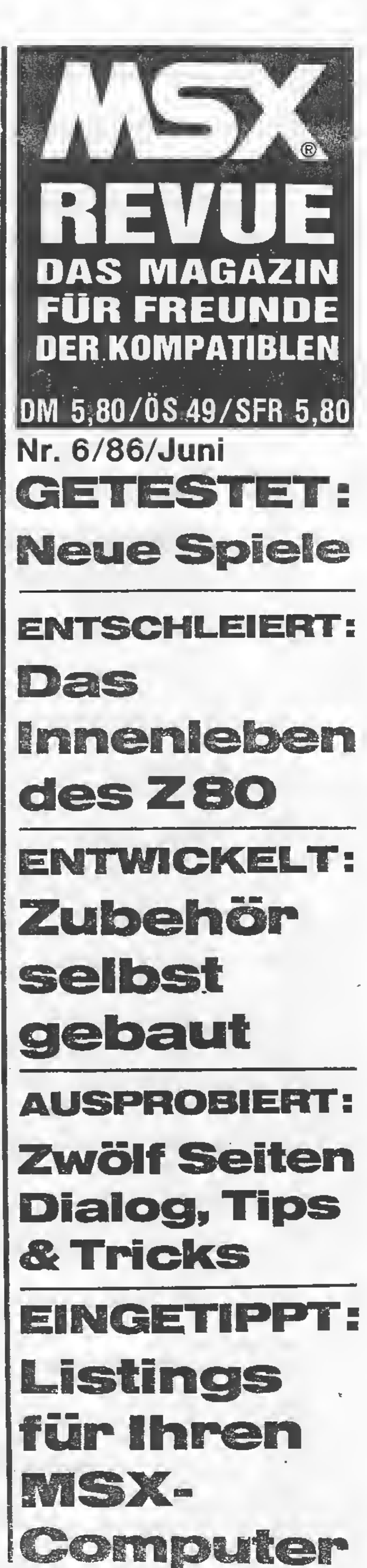
Wie Sie sehen können, sind auf Screen 5 noch 5 Zeilen frei für Ihre eigenen Meldungen.

BASE? (———) gibt auf dem Bildschirm die hiesige Basis an, womit gerechnet wird. Es ist in Forth nämlich sehr einfach mit einer anderen Zahlenbasis zu arbeiten, man braucht nur die gewünschte Basis in die Systemvariable BASE zu schreiben, also mit [8 BASE !] kann man octal rechnen.
BASE? speichert die hiesige Basis auf dem Datenstapel und gibt den Inhalt von BASE dezimal auf dem Bildschirm aus, nachdem erst Meldung 14 ausgegeben ist. Am Ende wird die alte Basis wieder hergestellt, da man sie eventuell zur dezimalen Ausgabe auf dem Bildschirm verändert hat.

FREE? (———) gibt auf dem Bildschirm die Anzahl der freien Bytes für Forthwörter und Datenstapel aus. Mit SP wird die untere Grenze des Datenstapels geholt und mit HERE die obere Grenze des Wörterstapels. Diese werden subtrahiert und das Ergebnis auf dem Bildschirm mit Meldung 13 ausgegeben. Weiter finden Sie auf Screen 3 noch zwei Wörter, die mit dem Datenstapel rechnen. SGN und .

SGN (n1———n2) gibt auf den Datenstapel das Zeichen von n1. -1 für negativ, 0 für 0 und 1 für positiv. Da eine Dividierung durch 0 in TI-Forth 0 ergibt, ist die Definierung des Befehls sehr einfach. (n1 n2———n3) liefert auf den Datenstapel n1 zur Potenz n2. Das Wort präpariert den Datenstapel so, daß in der Schleife immer n1 auf der zweiten Position des Datenstapels steht, man braucht dann diese Zahl nur zu holen und zu multiplizieren, nach der Schleife wird sie dann gelöscht.
Das Wort .S haben wir schon kennengelernt.
Es gibt die Zahlen, die im Datenstapel stehen, auf den Bildschirm ohne den Datenstapel aus, ohne den Datenstapel zu ändern. Beim Entwickeln von eigenen Forthwörtern ist es sehr nützlich, wenn so was auch mit dem Returnstapel gehen würde.
Diesem Zweck dient das Wort .RS (Screen 6).
Es gibt alle Returnadressen und Zwischenwerte, die auf dem Returnstapel stehen, auf dem Bildschirm aus. Für den Wörterstapel gibt es auf dem -DUMP Screen ein ähnliches Wort, VLIST.

Rob Prikanowski



LISTING

```
SCR #1
( DEPTH, PICK und ROLL )
: DEPTH
   SØ  SP - 2 / 1- . ;
: PICK
   DUP + SP +  ;
: ROLL
   1+ DUP DUP + SP + DUP  ROT ROT 2- SWAP
   Ø DO
     DUP DUP  SWAP 2+ ! 2-
   LOOP
   DROP DROP ;

SCR #2
( PICK und ROLL mit Fehleranzeige )
: äPICKü   DUP + SP +  ;
: PICK
   DUP DEPTH 2- SWAP - Ø< 5 ?ERROR DUP Ø> NOT 5 ?ERROR äPICKü ;
: äROLLü   1+ DUP DUP + SP + DUP  ROT ROT 2- SWAP
           Ø DO
             DUP DUP  SWAP 2+ ! 2-
           LOOP
           DROP DROP ;
: ROLL
   DUP DEPTH 2- SWAP - Ø< 5 ?ERROR DUP
   1 > NOT 5 ?ERROR äROLLü ;

SCR #3
( ?BASE, FREE, SGN und    )   BASE->R DECIMAL
: ?BASE
   BASE  DUP DECIMAL CR 14 MESSAGE SPACE . BASE ! ;
: FREE
   SP HERE - . 13 MESSAGE ;
: SGN
   DUP ABS / ;
:
   DUP Ø= IF   DROP DROP 1
          ELSE SWAP 1 ROT Ø DO OVER * LOOP SWAP DROP ENDIF ;
R->BASE
```

LISTING

```
SCR #4
( ERROR MESSAGES )
empty stack
dictionary full
has incorrect address mode
isn't unique.
bad argument
disk error
full stack
illegale index
file i/o error
floating point error
disk fence violation
can't load from screen zero
bytes free
base =
TI FORTH --- a fig-FORTH extension

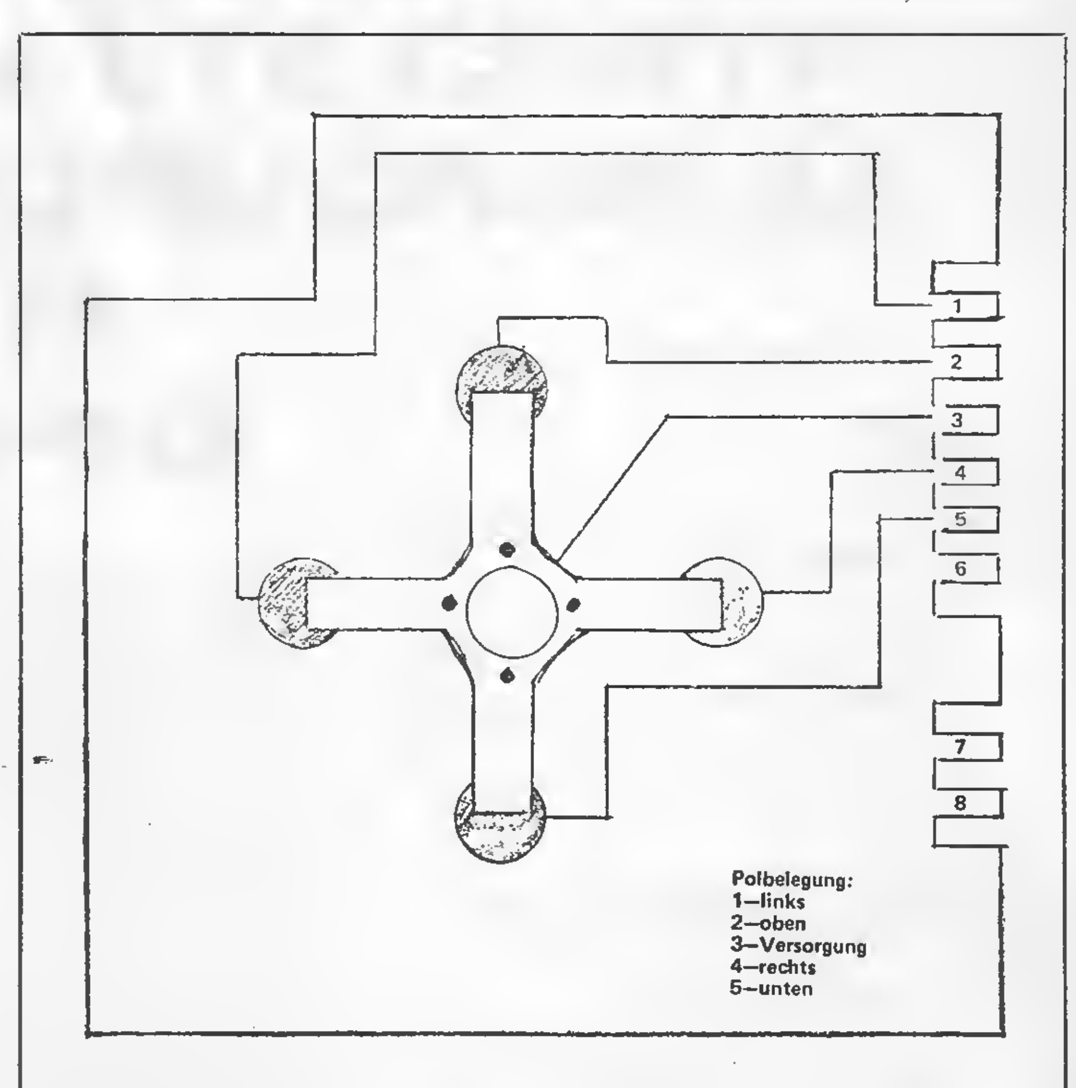
SCR #5
( ERROR MESSAGES )
compilation only, use in definition
execution only
conditionals not paired
definition not finished
in protected dictionary
use only when loading
off current editing screen
declare vocabulary
bad jump token
string to long

SCR #6
( Return stack printer by R.H.H.Prikanowski )
BASE->R HEX
: .RS
      CR 831C  2- RØ  2- ." Rö " OVER OVER = Ø=
      IF
            DO
                I  U. -2
            +LOOP
      ELSE
            DROP DROP
      ENDIF ;
R->BASE
```

JOYSTICK AM TI OHNE ADAPTER

Mit umgebautem Joystick direkt an den TI

Viele Joysticks konnte man bisher nur über einen Adapter an den TI 99/4A anschließen. Dem soll nun abgeholfen werden. Diese Beschreibung zeigt den Umbau des Quickshot II für den TI.
Als erstes löst man die drei Schrauben am Griff des Joysticks. Danach nimmt man die Griffschalen ab. Dann die Feuerknöpfe abziehen und die beiden kleinen Platinen herausnehmen.
Als zweiten Schritt löst man die vier Schrauben der Bodenplatte. Dann nimmt man das Oberteil ab und führt die beiden Platinen durch die mittlere Öffnung. Danach nimmt man die große Platine heraus und zieht die Stecker an der rechten Seite ab. Man zieht das alte Kabel raus und ersetzt es am besten durch ein flexibles 6poliges Kabel. In Bild (1) sieht man die Polung der neu anzubringenden Litzen. Für die große Platine benötigt man 5 Litzen, die 6. ist für die beiden kleinen Platinen bestimmt. Nun sucht man sich für jeden der fünf Anschlüsse eine Farbe aus und merkt sie sich für den Stecker. Danach werden an den Anschlüssen 1–5 die jeweiligen Litzen angelötet. Wenn das geschehen ist, lötet man die beiden Litzen der kleinen Platinen ab. Sie werden ersetzt durch die 6. Litze und durch ein kleines Kabel, das an Pol 3 angelötet wird. Diese beiden Litzen führt man nun durch den Griffstummel, Bild (2), unter der großen Platine vorbei zu den Anschlüssen und dem 6poligen Kabel.
Dann setzt man das Oberteil der Grundplatte wieder auf und schraubt es fest. Danach setzt man die linke Seite des Griffes an den Stummel und baut die kleinen Platinen wieder ein. Nun werden die Feuerknöpfe angebracht und die andere Seite des Griffes festgeschraubt.
Als Zugentlastung des herausführenden Kabels ist eine Stecknadel bestens geeignet. Man steckt sie einfach von innen durch das Kabel.



AUF DIE NUMMERN ACHTEN!

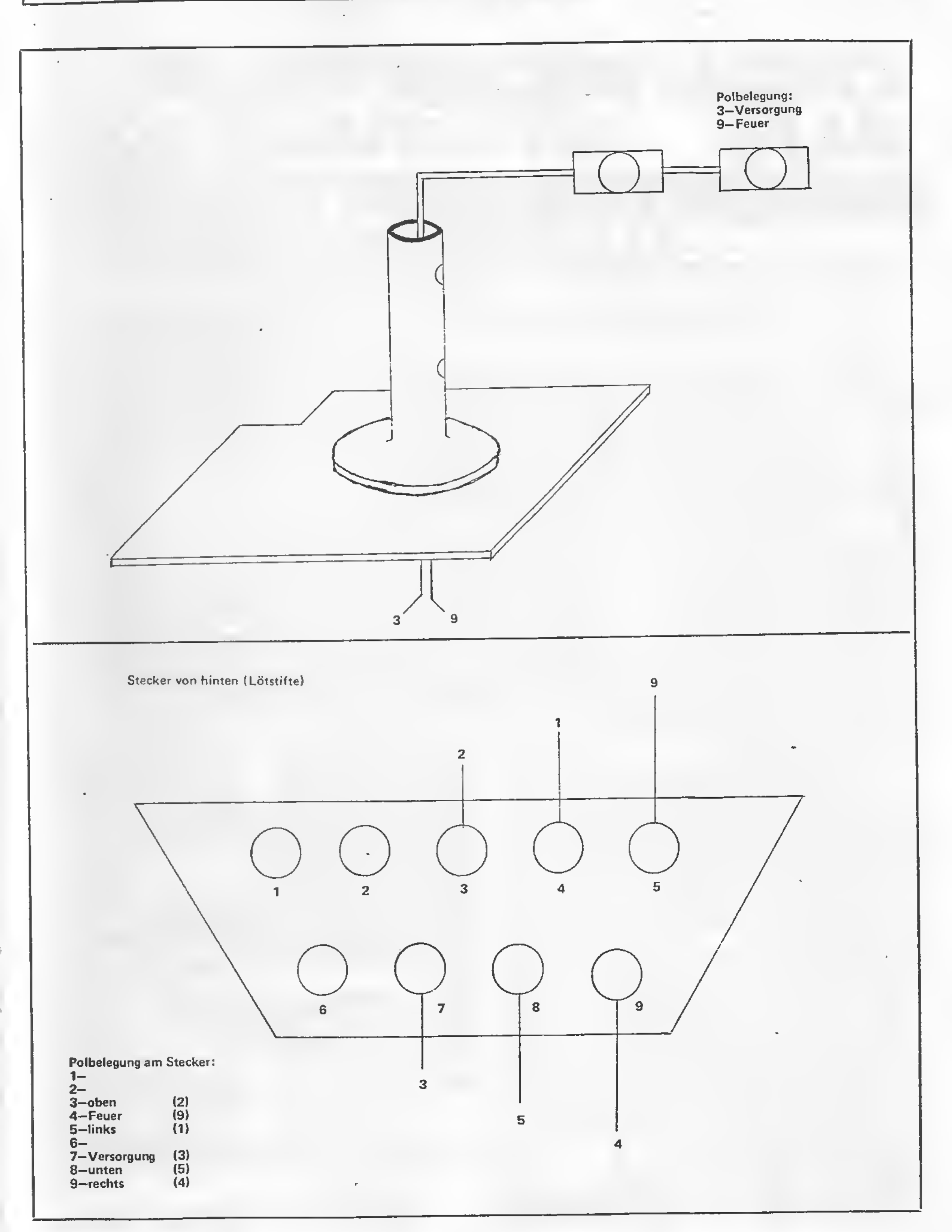
Wenn man den Stecker so vor sich liegen hat, daß die Lötstifte oben sind, kann man kleine Nummern erkennen. Nummer 1 und 2 sind nicht belegt, an Pol 3 kommt Nr. 2 des Joysticks, an Pol 4 Nr. 9 und an Pol 5 die Litze Nr. 1. Nummer 6 ist wieder nicht belegt und an Pol 7 kommt Litze Nr. 3, an Pol 8 Nr. 5 und an Pol 9 die Litze Nr. 4.
Am elegantesten ist es, wenn man den 9poligen Stecker mit einem Steckgehäuse versieht.

Michael Brakhage

STÜCKLISTE:
1 Joystick Quickshot II (wenn noch nicht vorhanden)
1 6poliges Kabel
1 9poliger Stecker
Die Preise für alles liegen zwischen DM 30,— (Conrad) und 50,— (Elektronikfachhandel)

Anm. d. Red.: Diese Art der Joystick-Anpassung funktioniert am TI 99/4A nur, solange nur ein Joystick verwendet wird. Beim Anschluß von 2 Stück beeinflussen diese sich gegenseitig.

TIPS & TRICKS



Polbelegung:
3–Versorgung
9–Feuer
3
9
Stecker von hinten (Lötstifte)
9
1
2
1
2
3
4
5
6
7
8
9
3
5
4
Polbelegung am Stecker:
1–
2–
3–oben (2)
4–Feuer (9)
5–links (1)
6–
7–Versorgung (3)
8–unten (5)
9–rechts (4)

CALL FILES AUS ASSEMBLER

In letzter Zeit erreichten mich einige Anfragen, wie CALL FILES aus Assembler her angesprochen werden kann. Richtig ist, daß dieses, wie so einiges Interessante mehr, leider in keinem Buch von Texas Instruments beschrieben worden ist. Dennoch ist es durchaus möglich, wir werden noch sehen, daß dies ganz einfach ist.

CALL FILES wird häufig bei Assembler-Routinen, die vom Basic her aufgerufen werden, benutzt, um Platz im VDP-RAM zu reservieren. Dennoch möchte ich eigentlich vor dieser Anwendung warnen, denn dieser Platz wird ja nicht für das Assembler-Programm, sondern für den Disk-Controller reserviert. In unserem Assembler-Sonderheft ist da im Rahmen der „Soundliste für X-Basic" eine viel bessere Methode beschrieben worden.
Nun kann aber natürlich CALL FILES in einem Assembler-Programm benutzt werden, wenn tatsächlich mehr als drei Dateien gleichzeitig auf den Diskettenlaufwerken eröffnet werden müssen.
Das CALL deutet schon darauf hin, daß es sich dabei um ein Unterprogramm handelt. In diesem Fall liegt es im ROM des Disk-Controllers und wird damit ganz „normal" über DSRLNK aufgerufen, nur, anstelle des DATA 8 brauchen wir ein DATA 10. Das DSRLNK braucht nun üblicherweise noch einen Peripheral Access Block (PAB), der eine Menge Informationen enthält. Von diesem ist bei einem Unterprogramm aber nur der Name nötig, alle anderen Angaben können wegfallen. Was sollte hier auch z.B. irgendein Opcode.
Dafür braucht das Unterprogramm natürlich einige andere Werte, in unserem Fall bei CALL FILES die Anzahl der Files. Diese wird auf der Speicherstelle >834C, also im FAC, übergeben. Nun wäre es prinzipiell möglich, das Unterprogramm mit dem Namen FILES aufzurufen. Das hat aber einen entscheidenden Fehler. Dieses sucht sich seine ganzen Angaben aus dem VDP-RAM aufgrund der gesamten vom Basic errichteten Pointer heraus. Texas Instruments hat für unseren Fall ein besonderes Unterprogramm im Disk-Controller installiert, welches auf den intelligenten „Namen" >16 (ein Byte) hört. Damit wurde sichergestellt, daß dieses Programm auf gar keinen Fall aus dem Basic aufgerufen werden kann, denn der ASCII-Wert >16 ist ja über die Tastatur nicht erreichbar.
Das Beispiel zeigt nun, wie das Unterprogramm aufgerufen wird.
Zuerst wird der Name des Unterprogramms in das VDP-RAM geschrieben. Hier ziemlich umständlich, aber das soll der besseren Deutlichkeit dienen. Es wird also >0116 in das VDP-RAM; gelegt. Dann wird auf >836C (Byte!) die Anzahl gelegt, in unserem Beispiel also 1, jedoch ist jeder Wert zwischen 1 und 9 (bei manchen Disk-Controllern auch 10 gleich >0A möglich). Der hier verwendete Befehl MOV könnte damit auch MOVB heißen.

CALL FILES DARF NUR AM ANFANG EINES PROGRAMMES DURCHGEFÜHRT WERDEN

Es folgt die übliche Pointerbelegung von >8356 für DSRLNK. Dann kommen noch ERROR-Abfragen. Das JEQ ERROR wird ausgeführt, wenn das Unterprogramm nicht gefunden wurde, wenn also kein Disk-Controller angeschlossen ist. Trat während der Ausführung des Unterprogramms ein Fehler auf, dies ist zum Beispiel der Fall, wenn die Anzahl für die Dateien zu groß gewählt wurde, ist die Speicherstelle >8350 ungleich 0. Dieser Error wird durch ein Lesen von >8350 in ein Register und dem anschließenden JNE abgefragt.
Das war dann schon das ganze Hexenwerk. CALL FILES darf aber immer nur zu Beginn eines Programmablaufes durchgeführt werden. Es ändert den Bereich des frei zur Verfügung stehenden VDP-RAMs. Der Zeiger auf die höchste freie Speicherstelle im VDP-RAM steht auf >8370. Auf gar keinen Fall darf CALL FILES von einem Assembler-Programm aufgerufen werden, welches von einem Basic-Programm gestartet wird und auch wieder zu einem Basic-Programm zurückkehrt.
Auch müssen, bevor CALL FILES aufgerufen wird, alle Dateien auf Disketten vorher geschlossen sein. Wird dies nicht beachtet, kommt es unweigerlich zu Fehlfunktionen des

```
FILES   BYTE >Ø1         *LAENGE DES NAMENS
        BYTE >16         *NAME DES UNTERPROGRAMMS

SUBP    LI   Ø,>Ø68Ø     *CALL FILES
        LI   1,FILES     *MIT SUB >16
        LI   2,>ØØØ2
        BLWP @VMBW       *PAB NAME SCHREIBEN
        LI   1,>Ø1ØØ     *IM ERSTEN BYTE ANZAHL FILES
        MOV  1,@>834C    *ANZAHL SETZEN
        MOV  Ø,@>8356    *POINTER FUER DSRLNK BELEGEN
        BLWP @DSRLNK
        DATA 1Ø          *SUBPROGRAMM AUFRUFEN
        JEQ  ERROR
        MOV  @>835Ø,1    *ERROR ABFRAGEN
        JNE  ERROR

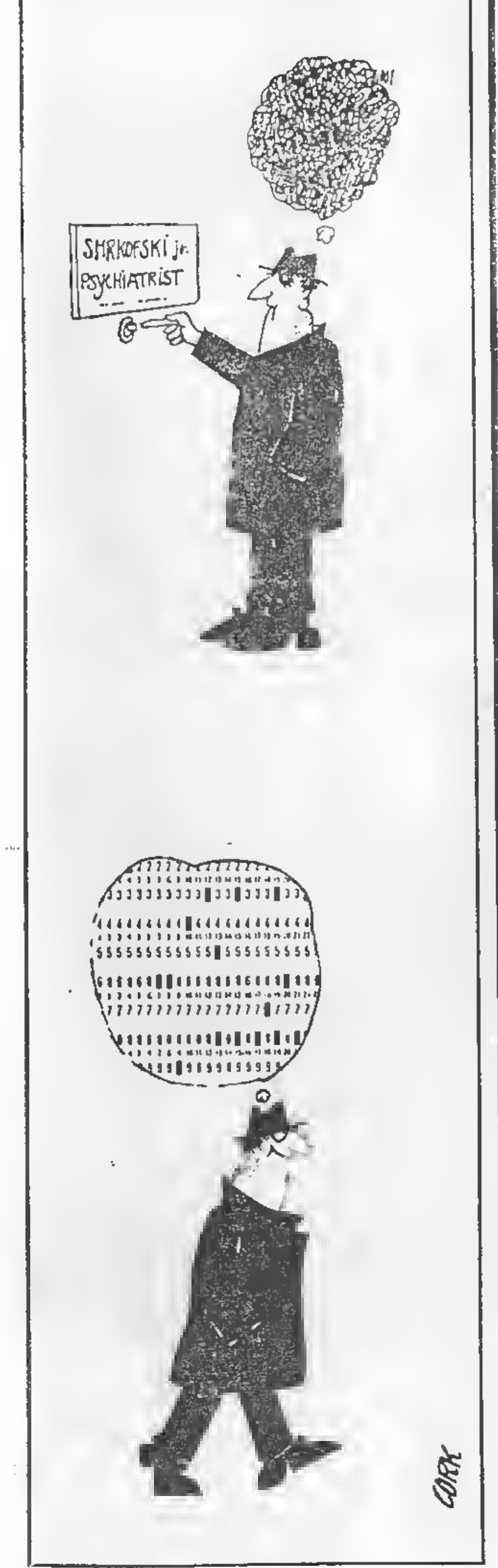
* NUN HIER PROGRAMM
  WEITER

ERROR  LI   Ø,>3ØØØ   * HIER ERRORHANDLUNG EINFUEGEN
       MOV  Ø,@>837Ø  * Z.B. FIKTIVER WERT ENDE DES RAMS

       END
```

Basic-Programms bzw. des Disk-Controllers, welches Datenverlust und sogar die „Zerstörung" der eingelegten Disketten zur Folge haben kann. Unter Zerstörung ist dabei natürlich nicht eine mechanische Beschädigung gemeint, sondern der Verlust aller auf der Diskette befindlichen Daten/Programme.

Heiner Martin



VIEL SPASS MIT APESOFT-GRAFIK, ABER ...



Viele TI-User haben mittlerweile die Apesoftgrafik durch das neue Extended Basic II Plus kennengelernt und haben viel Spaß mit ihr. Leider gibt es jedoch für das Apesoft-Basic nicht sehr viele Programme, die helfen, Befehle zu verstehen.
Mit diesem Bericht möchte ich dazu aufmuntern, sich mal wieder mit der wirklich schönen Grafik von Apesoft zu beschäftigen. Einige Beispielprogramme sollen Anregungen für weitere neue Ideen mit Apesoft geben.
Ein sehr interessanter Befehl des Apesoft ist "TURN", mit dem sich schöne Grafiken erstellen lassen. wie dieses erste kurze Programm zeigt:

```
100 REM TURN&RECT
110 REM +++++++++
120 REM
130 REM MATTHIAS ORF
140 REM
150 REM
160 CALL LINK("GRAFIC".0)
170 CALL LINK("WINDOW",3.8)
180 CALL LINK("SETTO",65,60)
190 FOR A=2 TO 40 STEP 2
200 B=A+5
210 CALL LINK("RECT",A,B,-A,
    B,A,-B,-A,-B)
220 CALL LINK("TURN",9)
230 NEXT A
240 GOTO 240
```

Bei diesem Programm wird ein Rechteck, das aus vier Einzelrechtecken besteht, nach jeder Ausgabe auf dem Bildschirm um neun Grad gedreht und wieder neu gedreht. Hierbei entsteht ein interessantes Muster. Die Zahl in Zeile 220 hinter TURN sollte man einmal ändern, und schnell erhält man ein anderes Muster.
Im nachfolgenden kurzen Programm dreht sich mittels TURN ein Quadrat um 89 Grad nach links, wobei ein schönes Muster entsteht. Hier sollte man auch einmal die Variablen in den Zeilen 190 und 210 ändern, um andere Muster zu erhalten!

```
100 CALL LINK("GRAFIC",0)
110 CALL LINK("WINDOW",3,8)
120 CALL LINK("SETTO",64,60)
130 FOR I=5 TO 92
140 CALL LINK("MOVE",I)
150 CALL LINK("TURN",89)
160 NEXT I
170 GOTO 170
```

Ändert man bei diesem Programm in Zeile 150 die Zahl 89 in 170 um, so entsteht ein wunderschöner Stern. Mittels einer Umänderung kann man so mit Apesoft eine vollkommen neue und andere Grafik erhalten.
Mit dem letzten Programm werden die schönen Farben mit Apesoftgrafik zusammen präsentiert. Es werden in immer neuen Farben, die mittels Zufallsgenerator gewählt werden, Säulen dargestellt. Drückt man während des Programmablaufs die Leertaste, so endet das Programm, ansonsten wird, wenn fünf Säulen auf dem Bildschirm dargestellt sind, der Screen gelöscht und es werden fünf neue Säulen dargestellt.
Man könnte in das Programm beispielsweise einbauen, welche Farben für Vorder-, und Hintergrund gewünscht werden, oder man könnte das Programm total umkrempeln und für jede Säule einen Wert eingeben lassen; es sind der Phantasie also keine Grenzen gesetzt.
Zeile 260 sollte man einmal vergrößern oder verkleinern. Wird die Zahl verkleinert, so stehen die Säulen dichter nebeneinander, wird die Zahl vergrößert, so stehen die Säulen weit auseinander.

```
100 RANDOMIZE
110 CALL LINK("GRAFIC",0)
120 CALL LINK("WINDOW",3,8)
130 CALL LINK("CENTRE",1,80)
140 E=0
150 A=INT(RND+100)+1
160 V=INT(RND+13)+3
170 H=INT(RND+13)+3
180 CALL LINK("SETCOL",V,H)
190 CALL LINK("HSTDIA",-X,
    -35,10,A,12)
200 X=X-20
210 IF X=-100 THEN CALL LINK
    ("CLTBLE"):: X=0 :: GOTO
    220 ELSE 220
220 CALL KEY(O,K,S)::IF K=32
    THEN 230 ELSE IF K 32
    THEN 140
230 CALL LINK(CLTBLE")::CALL
    CLEAR :: END
```

Vielleicht macht mal jemand aus diesen und anderen Grafikbeispielen ein großes Apesoftgrafik-Demo und schickt dieses als Programm an die TI-REVUE! Alle Apesoftbesitzer würden sich bestimmt freuen, neues "Futter" für diese Grafikerweiterung zu bekommen.
Wer noch an weiteren Programmen für Apesoftbasic Interesse hat, der kann von mir gerne noch weitere Programme erhalten.
Ich freue mich über jeden Brief, egal ob positiv oder negativ zu diesem Bericht.

Matthias Orf

BÖRSE

Suche billig TI-Marketing Modul, C. Monzel, Tel: 0208/35389.

Verk. TI 99/4A + Ext. Basic 3 32K-Erweiterung. Tel: 06071/48198.

Verk. TI 99/4 A + Ex.Basic + 2 TI-Joysticks + Rec. Kabel + 30 Progr. + Zeitschrift. für 250,— DM. Tel: 02445/8023.

Verk. TI 99/4 A, Centr. Interf., 9 Module (ExTB Diagnose, Schach), 2 Joysticks, viel Literatur, VB 750,— DM, Tel: 02734/60447.

TI 99/4 A, Ex Basic, Rec.Kabel, Joystick + Adapter + Modul Datenverw., div. Progr. Literatur. 450'- DM. Tel: 0202/626993.

Wer will Spiele tauschen? Wollen Sie eine Modultestzeitschrift. Dann rufen Sie ab 18.00 Uhr die Nummer: 05231/66544.

Wer tauscht eines seiner Module gegen 25 Superspiele, wie Slicks oder Moon Race, ab 18.00 Uhr, Tel: 05231/66544.

Suche TI 32 K-RAM-Karte für P-Box. W. Rexin, Hauptstr. 135, 4800 Bielefeld 14.

Verk. TI 99/4 A, Ex-Basic, Box, 32 K-RAM, RS232, Disk-Contr. + Disk-Manager, + 1 Laufw. + 2 Module: Text u. Datei, Datenv. + Analyse. Tel: 0231/871581. Preis: VB.

Verk. Peri-Box, Laufw., Contr., Manager 1050,— DM, Ex.-Bas. 130,— DM, Soccer 30,— DM, Joystick 40,— DM, Tel: 05121/62716.

Verk. Ext. 32 K Erweiterung (batteriegepuffert) und orig. Ex-Basic für 400,— DM. Sven Porstendorfer, Brahmsstr. 23 2200 Elmshorn.

Suche für TI-Artist und für Graphik. Hardcopy für Seikosha. GP-700 A und TI 99/4 A. Tel: 0234/353958.

Verk. Zeitlupenmodul (J/O Port) mit Standbild und Geschwindigkeitsregelung für 60,— DM. Sven Porstendorfer, Brahmsstr. 23, 2200 Elmshorn.

Suche ext. o. internen Diskcontr. Michael Buresch, Am Freidhof 23, 2200 Elmshorn.

Verk. ZIC Module u.a. MM, MRS, Pacman, Pole, Position, Preis? 32 RAM = 100,— DM. Tel: 02407/8911, ab 19.00 Uhr.

Verk. TI 99/4 A + Zubeh. z.B.: RS232, 32 K RAM, Peri-Box, Ed, As., Ext. Basic. Tel: 0911/512793.

Verk. TI 99/4 A + Ex.Basic + Joystick + Zubehör + Cass.-Rec. Alles nur 300,— DM + Hefte. Tel: 089/6377807.

Verk. Original TI Disc-Contr. karte für Peri-Box 200,— DM M. Freuler, Crütstr. 58, CH-8625 Gossau, Tel: 0193/54252.

!!!HILFE!!!
Benötige dringend Ex.Basic-Handbuch (deutsch) evtl. im Tausche gegen Minus-Mission-Modul. Angebote an: Malte Stief, Tel: 0421/832194.

Suche X-Basic mit dt. Handbuch (ca. 100,— DM). Tel: 08142/14737.

Suche günstig Peri-Box mit Anschlußkabel, Controller, Disk, Disk Man. Tel: 07532/8011448, ab 16.00 Uhr, Klaus Kromer.

Commodore VC 20 (neu), Netzteil, Videointerf. und Literatur, 150,— DM Olympia Kofferschreibmaschine 100,— DM. Max Markl, Vorderstraße 11, 7240 Königstein 3, Tel: 06173/1588.

Suche Diskkontr. für doppelseitiges Laufwerk oder Schaltplan (Copy). Tel: 0751/45981, ab 18.00 Uhr.

TI-Anfänger sucht kostenlos Programme in Ex.Basic (alles) oder gegen ger. Beträge auf Kass. Christian Monzel, Hingbergstr. 128, 4330 Mülheim-Ruhr.

Verk. RS232/Forth/Buch TI Intern/TI Rechnungsstellung/ TI Adv.-Spiele/ Tausche Progr./Mod. Zero Zap. Tel: 07141/461147.

Verk. Exbas. Handbuch deutsch/englisch zus. 50,— DM. TI-Rev. Ausg. 1/84 - 4/86 zus. 30,— DM. Tel: 089/3137650.

Suche: TI-Kontakt im Raum Donauwörth. Ex-Basic + E/A + Minimem. 32 K + Disk + RS. E. Hilger, Tel: 0906/91676.

Verk. Schachmodul für 75,— DM. Michael Buresch, Am Freidhof 23, 2200 Elmshorn.

Suche Tauschpartner mit DSK für E/A- und Ex. Bas. Progr. P. van Eecke, Im Naurt 3, 6940 Weinheim.

Verk. TI-99/4 A wegen Systemwechsel. Info von: Markus Theimer, Schwedenstr. 2, 6203 Hochheim.

Verk. TI Literatur: Tips & Tricks 20,— DM, 99 Special I 25,— DM, 99 Special II 30,— DM, Chip Programmsammlung 10,— DM. Ausserdem einen Basic-Kurs auf 10 Kassetten. Verkauf auch einzeln. Tel: 0214/401384, 17.00 bis 20.00 Uhr.

Hallo TI-User! Das ist sensationell! Verk. Tunnel of Doom-Modul für 30,— DM. Tausche außerdem Spiele in TI-Basic und extended Basic. Verk. Marketing-Planspiel und Ihr Finanzberater (je 7,— DM) Also ran ans Telefon: Thorsten Löw, 07130/7165.

Verk. orig. Adventure-Modul (mit Pirate Adv.) 50,— DM. Norman Keller, Tel: 06081/7477.

Suche Editor/Assembler (90), dt. E/A Handbuch (35), Tel: 09741/3654, ab 17.00 Uhr.

Suche: Diskcontr. DS6DD intern oder extern. Sprachsynthesizer, Modulgehäuse, Programmiere Ihre Eproms: z.B. für Modulplatine, Q-Save-Modul. Angebote an und Info von: Peter Rieger, Talstr. 64, 6750 Kaiserslautern, Tel: 0631/73669.

Suche kaufm. Software, sowie E/A Modul + Software. Angebote bitte an: Tel: 091 23/12249, tägl. ab 18.00 Uhr.

Achtung TI 99 User
Ich möchte meine externe V. 24 (RS232) Schnittstelle von Texas Instruments für 250,— DM verkauf. Sie können mich wochentags von 15.00 bis 19.00 Uhr unter 08161/615 37 erreichen.

TI 99/4 A, Ext. B. + dt. Handbuch, Recorder + Rec.-Kabel, TI-Joysticks, 120 Progr., Literatur, für 330,— DM A. Hausendorf, Normannenstr. 61, 4200 Oberhausen 1. Tel: 0208/863619.

Endlich ist es da!!! Deutsches Handbuch für TI-Writer und TI-Forth. Preis pro Buch: 45,— DM. Ralph Weber, Langemarckstr. 54, 8906 Gersthofen.

TI 99/4 A + Ex. Bas. + Handbuch + Rec. Kabel + Adap., Joystick + Cetronics Schnittstelle 450,— DM. Tel: 024 08/58090.

Verk. TI 99/4 A. Sven von Loga, Königsforststr. 29, 5000 Köln 91 (Brück)

Verk. TI 99/4 A, 3 Module, Rec.Kabel, RS232-Schnittstelle, Drucker-GP 100 A, VB 520'— DM. Tel: 0281/2012128.

Verk. TI Grafiktablett, VB 150,— DM. Bei: Michael Haasler, Tel: 0711/766526.

Int. MSX-Club?!? Man, das wär doch was! Von Anfang an dabei sein! Infos und Progr. austauschen! Mit Clubzeitung und Archiv! Heute noch eintreten. Info für 2,— DM (bar) bei: Helge Jans: Hohe Luft 18; 3111 Suhlendorf.

Computer-Zeitschriften billig abzugeben, CPU, HC, Chip, Happy, CK, Comp.-Per, Total 140,— DM. Tel: 06157/4684.

Seikosha-Drucker 550 A mit Interface für TI 99/ 4 A, 18 Schriftarten, kompl. mit deutscher Anleitung 900,—DM VB — auch einzeln. Tel: 06142/562190.

Verk. TI 99/4A Komplettsystem. Kons. P.-Box 32K RS 232, orig. LW, 2 Slim-line, LW a 180 K, Ext. Basic, Ed-Ass, TI-Writer, 2 Joyst., Pac-Man, TI-Invaders, Software + Literatur. — Auch einzeln! H. Rögner, Lammsg. 12, 8500 Nürnberg.

Verk. wegen Systemwechsel TI 99/4 A mit viel Zubehör! Auch einzeln. Tel: 0211/422216 tägl. ab 18.00 Uhr.

Verk. TI 99/4 A + Recorder + RecKabel + Schachmodul + Spiele + Buch + Zeitschr. nur 300,— DM. N. Borsch, Schöner Blick 31, 8560 Lauf. Tel: 09123/12125.

Verk. TI 99/4 A Modulator, CH 4, 30,— DM. Netzteile 20,— DM. Trafo 20,— DM. Ed. Assembler, Buch 40,— DM, TI Intern 20,— DM. Tel: 06155/4692;

BÖRSE

TI 99 + TI Rec. + ExBas + Kabel für 2 Rec., Bücher, Hefte, 50 Stück und anderes Zubehör. Preis VB. Tel: 02041/28321.

Texas Disklaufw. Intern wenig benutzt 280,– DM, 35 Disk. voll, VB. Tel: 02041/28321.

Maschinensprache für Cass. Verkaufe Assembler + Disass. auf Eprom für den ROM-Bereich, Modulplatine mit TI E/A + Line by Line Ass. + Disass. im ROM-Bereich für 99,– DM, 32 K notw. Tel: 04421/203151.

Verk. TI 99/4 A + Ext. B. + Cass.Rec. + Kabel + TI Joystick + Prog.Cass. + viel Literatur. 400,– DM VB. Tel: 02234/14615.

Wer tauscht für TI-Ext.B. Programme? Melden bei Helmar Schuschnig. Tel: 089/8369 58.

Verkaufe Text-Editor m. Anleitung 50,– DM. Module: „Amazing" u. „Hustle", zus. 50,– DM. Tel: 040/6523901.

Verk. TI 99/4 A, Ext. B. + Joystick Adapt, ca. 200 Spiele (Cass.) 330,– DM. C. Traugott, Humboldtstr. 130, 8500 Nürnberg 40.

Verk. TI 99/4 A mit dt. Anleitung + Ext.B. + TI-Invader + Parsec + Joystick, 400,– DM. Tel: 0911/745157, ab 14.00 Uhr.

Verk. C 16 + 64 K + Datas. + 2 Joystick Adapter + Turbo-Tape + 5 Bücher + viele Programme für 400,– DM. Tel: 02445/8023.

Assembler-Programme für E/A, XB, MM + 32 K: Disk-Editor, Fast-Copy, GPL-Disassembler. Info gg. Freiumschlag. Alles sofort lieferbar! M. Eichhorn, Ziegelheck 1, 6240 Königstein 4.

Verkaufe TI wegen Hobbyaufgabe. Habe alles: 2 Laufwerke, P-Code, Ed. Assemb., alle Module, Sprachein-, ausgabe, Matrixdrucker und vieles andere. Rufen Sie 07174/5206, ich schicke per Post Aufstellung. Auch Einzelkauf möglich.

Verk. div. Module z.B.: Schach 50,– DM, TE II 60,– DM, Statistik 30,– DM u.v.m. - auch Progr. auf Dsk. o. Cass. Tel: 09223/12249.

Verk. Drucker, Epson RX 80 m. Interf. anschluß f. TI 99 40,– DM, 600,– DM TI-Thermodrucker, 200,– DM Datenverw.-Module, Text und Datei, je 50,– DM, Tel: 04621/33294.

Habe noch Module original verp. Ex 130,– DM, Munch Man 30,– DM, Yahtzee 20,– DM, Personal-Rep. 50,– DM, Datenverw. 30,– DM Or. Joy. Arno Heinrichs, Hauptstr. 72 a, 5068 Odenthal, Tel: 021 74/40616.

Hallo TI-Freaks!!!
Verk. tausche Software.
Verk. Extened Basic. Tel: 0201/712816.

HIRES GRAPHIK für X-Basic (Sinuskurven, Kreise u.v.a.) 10,– DM (inkl. Kassette u. Porto, Schein) an: Julian Pyc, Schönblick 7, 8050 Freising, Tel: 08161/62165 Prog. mit ausf. Anleitung. XB-Unterprogramme.

Suche Schaltplan vom TI 99/4 A. Verk. geg. Höchstgeb. Schachmodul. Tel: 0208/608869.

Suche/Biete + Tausche TI 99/4A Hardware + Original Software, G. Bürger, Im Pump 5, 3051 Auhagen, ab 17 Uhr Tel. 05725/6409

ACHTUNG TI-USER!!!!!!
Da ich unbedingt Geld benötige, bin ich gezwungen, ein paar Zubehörteile meines TI.s zu verkaufen: Das wären das Household Budget Management
Modul Preis VB
Yahtzee-Mdoul Preis VB
Zero Zap-Modul Preis VB
Beginning Grammers-Modul (ideal für Englischanfänger) Preis VB
Meldet euch unter 07156/349 41 JAN

Dringend. Suche das Modul möglichst billig!!!
07156/34941 JAN

Wer hat Interesse, mit mir E/A-Programme (hauptsächlich Spiele) zu tauschen? Ralph Benzinger, Wilhelmstr. 65, 6800 Mannheim 51

Invaders (Modul) 30,–; Parsec (Modul) 35,–; Tombstone (Modul) 25,–; Schachmeister (Modul) 60,–; Superspiele (Buch 15,– jewils + Porto (Nachnahme o. Vorauskasse) T. 06245/6315

Verkaufe: Ti 99/4A 100,–
MODULE
Return to Pirate's 30,–
Adventure + 11 Adv. 70,–
Microsurgeon 30,–
Espial 35,–
Miner 2049 30,–
Dragon Mix 5,–
EX Basic 130,–
Moon Mine 22,–
HARDWARE
32 K-Erw. extern (CMOS-RAM) 150,–
Joystick-Adapter 12,–
Sprach-Synthesizer 82,–
neuwert. Recorder + Recorder-Kabel 90,–
30 Kassetten (600 P.) 45,–
oder pro Stück 2,–
diverse TI-Bücher 95,–
diverse TI-Zeitschr. 45,–
TI Spezial 1 u. 2 10,–
Tunnels of Doom 37,–
Turbo Assembler (C) 37,–
bei Komplett-Abnahme ist der Preis 875 DM statt 1055 DM
Bestellungen an:
A. Schlaht/Große Str. 79/
2802 Ottersberg I
Tel. 04205/1857
Mo–Fr 18.30-20.00 h

Verk. TI99/4A, X-Basic+ Handb., 32 KB+Centronics+ Druck.Kabel Datenverw.+ Analyse, Text u. Da e/A, div. Literatur, 31 Taps m. Programm. Rec.Kabel + Joystick, wegen Aufgabe alles zusammen nur: 650,– DM Tel. 0941/88288 ab 17 h

Ti 99/4A Komplettsystem incl. Ex Basic, Box, Disk, 32K, RS 232 Software, diverse Module Orig. TI-Software (Writer, Lagerverw., Rechnungsstellung)
Div. Literatur (Special I + II, etc.)
M. Mayr, Adolf-Damaschke-Str. 7, 6102 Pfungstadt
VHB 2200,– DM

TI 99/4A, Kassrec., Exp. Box Laufw. (DSDD) + Contr. (4 Laufw) + Manager, 32K, RS232, ExBas. Datenv. + Analyse, Stat., Joyst., TI-Text/Datei, ID-Data/ID-Konto, Lagerverw., teilwe. Neugeräte, kompl. VHB 2.500,– evtl. einzeln, 05745/2181

Neue Software für Yamaha CS5, M: FM-Voicebankloader (3 Voicebanks in 1 mischen, 192 Sounds ohne M-Macro anspielbar), FM-Sequencechain (Composersongskoppeln), DX21-Simulator, Topsounds. Infocassette mit 48 Voices, 3,20 DM bei Frank-Dieter Klefe, Reichenauerstr. 5, 7900 Ulm-Lehr

Verk. od. Tausch! Homefinanc. Dec. Houshold-Management + Carwars + Alpinter + Tombstone-City + Musik-Maker + A-Mazeing + Indoor-Soccer + alle origin. TI-Module + Mini-Assembler v. Fa. Radix + 60 Comp. Kurs in 6 SM
** Su. Hard + Software od. sehr billig. *** Tel. Nr. 08841/40346********

Suche die Steuerkarte die in der B-Box neben dem Trafo und Gebläse sitzt und die Stromverteilung für die Steckkarten und das Laufwerk reguliert. Bitte anrufen telf. 0711/843775

*WORTSCHATZ * Das TI-Vokabel-Programm, das wirklich alles bringt. (ExBasci)
2 Disketten inkl. 3000 Vokabeln Englisch, nach Themen geordnet: DM 30 (Scheck od. NN) Info 1 DM od. Tel.: 0221 256696
Ulrich Wolf Fahrenheitstr. 16, 5300 Bonn 1

K K 000 N N TTT AAA K K TTT
KK 0 0 N N T A A K K T
KK 0 0 NNN T AAA K K T
K K 000 N N T A A K K T
XB+32K yeah!
U. Bolte, Am Sportplatz
3065 Nienstaedt 1

SUPERANGEBOT
TI99/4A + Compact Peripherie System CPS99 + Seikosha 500 Drucker + E/A-Packet (Modul, Buch, Disk) + EX-Basic + Schachmodul + sonst. Zubehör + alle Ti-Revue + Literatur + ca. 300 Prgs. * nur 3000,–
Tel. 06821/5531

Wissen Sie zufällig von der Existenz eines Kassetten-Interfaces für den CC-40 (evtl. von Privatanbietern oder TI-Clubs)? Da der CC-40 in vielerlei Hinsicht Software-kompatibel zu m TI 99/4a + ExBasic ist, wäre ich Ihnen dankbar, wenn Sie in Ihrer Zeitschrift auf dieses Gerät vielleicht einmal eingehen würden.
Stefan Ostendorf, Gasselstiege 57, 4400 Münster

Suche TI-Writer, Multiplan verk. TI-Cas. Finanzberater, Spiele 1,– + Sp. 2, Marketing B – für Anfänger je DM 10,– Mudol-Pac-Man 50 DM, Konsole mit Kabel 130,– DM. Drucke Listings kostenlos – nur Rückporto von Disk oder CAS.
Tel. 06805/8393

BÖRSE

Verk. Ext. Basic 80,– DM. Mini Mem. 80,– DM. Konsole 110,– DM. Chr. Lang, Tel: 0871/61574.

Achtung TI-Freunde!!! Verkaufe eine komplette TI-Anlage. Ela-Minimem-RS232 usw. weit über 200 Programme. Neupreis ca. 6500 DM für nur VB 4000 DM (nur komplett) Tel. 06101/86860

Ich verkaufe:
Das Spielen, Lcr. Arb. Handbuch zu 20 DM
Ein Ext. Basic Handbuch (Englisch) zu 25 DM
und eine Kassette m. ca. 30 Spilen zu 15 DM
Tel.806525 oder evtl. 859115
Andreas Berkenbrock, 4600 Dortmund 16, Evinger Berg 8

Verk.: Datenrecorder & Kabel nur 60,– DM, Modul Munchman für 20,– DM, Schachmodul für nur 50,– DM, Progr.kassetten mit divers. EX-Basic-Progs. (Anwender & Spiele) je 8,– DM, D. Biehl Tel. 06821/5531

*** FAST GESCHENKT ***
TI99/4A, Compact Peripherie System CPS99 (DS/DD), Druck. Seikosha 500 GP, Ex-Basic, E/A-Paket, TI-Writer, TI-Artist, Handbücher, Zeitschriften, ca. 300 Programme VB 2200,– DM, Tel. 06821/5531

Verkaufe für den TI-99/A4
100 Progr. 40,– DM
200 Progr. 80,– DM
300 Progr. 120,– DM
EINZIGARTIG bei Manfred Schneider; Tel. 06874-6705 (ab 14.00 Uhr)

Verkauf3e für den TI-99/4A
Verkaufe 2 Atari Joysticks, – 1 Quick-ShotII und das Modul Parsec für nur 100,– DM Manfred Schneider, Nunkircher Str. 42a, 6648 Wadern-Büschfeld, Tel. 06874/6705

Tausche 300 Programme für den TI gegen eine 32K-Ram Karte für die Peripherie-Box. Anruf genügt !!! Manfred Schneider Tel. 0684/6705

Wer hat Lust einen Sprachsynthesizer gegen 300 Superprogramme zu tauschen ???
Meldet euch bei:
Manfred Schneider, Nunkircher.Str. 42 a, 6648 Wadern-Büschfeld, Tel. 06874/6705

Yamaha CX-5M Musikcomputer incl. gr. Tastatur YK-20 u. div. Zubehör zus. 990,– DM Tel. 02161/16054

Tausche 300 Programme gegen Diskkontrolerkarte für die Peripherie-Box ...
Manfred Schneider, Nunkircher Str. 42 a, 6648 Wadern-Büschfeld, Tel. 06874/6705

Verkaufe!!! „Zusammenfassung von Call Load u. Call Peek" Info gegen 80 Pf-Briefmarke
2 Disks mit Text 20,–
Ausdruck der Texte 10,–
bei Zusend. von 2 Disks 10,–
Schein o. Scheck an M. Kraus, Vaihinger Str. 64, 7132, Illingen, Tel. 07042/24415

Hallo TI-User!!! Suche Kontakkte zum Programmtausch. X-Basic u. EIA-Modul DFÜ Tel-Nr. bekannt
Tel. 07042/24415

TI 99 Anwenderprogramme in Ex Basic z.B. Schachlernprogr., Mathsassistant (über 40 Rechenoperationen), Raglisten und Wahlauswerter uvm. Info bei Dirk Junghans, Am Fort-Bichler 9, 6503 MZ-Kastel (auch Programmtausch)

Hallo TI-Utility-Anwender!! Verkaufe Widerstandsberechnungs-Programm (22K) mit Bauteilbeschreibung (8K) auf Disk nur 10,–
Schein o. Scheck an M. Kraus, Vaihinger Str. 64, 7132 Illingen, Tel. 07042/24415, Info geg. 80 Pf-Briefmarke

---

Suche günstig verschiedene
! MODULE !
Wer seine Module verkaufen will, Angebote bitte schicken an W. Röthlein, Obere Dorfstraße 12, 8833 Dollnstein 2

GHTUNG * ACHTUNG * AC
Suche Spiele für den TI99/4A (z.B. Donkey, Kong, Espial, Miner 2049, Fathom, Q-Bert usw.) sowie Sprachsynthesizer, TI-Maus und Grafiktablett (Super-Sketch)
Georg Staudacher, Westerstr. 5, 2943 Esens, Tel. 04971/2574

Achtung!!!
Das original Ext.-Bas. Turmspringen aus der HC/März 1986 (Summergamesvariante) jetzt vom Autor anfordern!! Gegen 10 DM im Briefumschlag sende ich es auf Kassette zu! – Super Grafik – Super Sound – anfordern bei:
H.H.T.S.S
Christoph Hinsch
Achtern Diek 5
2082 Uetersen!!!

Vekaufe
TI99/4A, Rec. Kabel, Recorder (m. Bandzählwerk), Joystick, abges. 10-er Tastatur, Bücher, Zeitschr., div. Programme, nur komplett 300,– 02353/10090 ab 18.30 Uhr

---

Verkaufe Tips & Tricks (Data Becker) wenig gebraucht für 15,– DM.
Suche alte TI-JOURNAL-HEFTE. Melden bei: Klaus Vietzke, Victor-Goll.-Str. 43, 5170 Jülich. T. 02461/51660

Verk. 99/4A Kons, Ex-Bas. + Handbuch (engl.), Mod: Chess, Munchman 2 Cass. Kab, 2 Joyst., Bas-Lerncas. Programmierhandb. dtsch für Bas + Ex-Bas., div. TI-Bücher, viel Lit. (TI-Revue, TI-Spez. usw.) alles orig. TI alles zusammen für DM 500,– VB. T. 040/2700500 ab 19 h
Schmidt Gerd-Günter
Hölderlinsallee 6
2000 Hamburg 60

Verkaufe wegen Systemwechsels TI + P-Box + Disklw + 32 K + RS 232 + Centronics + Farbdrucker Seikosha GP 700 A + Ex Basic II + + E/A Modul + Speech Synt. + 5 Module + ca. 550 Prgms. auf ca. 60 Disks + Joysticks u. Gass. Kabel + Extras * VB 3200 DM * Interessenten melden sich bei Stefan Landgrebe, Schulstr. 8, 2915 Ramsloh, Tel. 04498/440

Verkaufe TI 99/4A + Recorder + Ext. Bas + Miniem + Buch(D) + Schach + Car Race + Lit. + Csss. Inmann Werneria – 4040 Linz, Altenbergerstr. 74, 5500 ÖS Tel. 0732/236431/348

Verkaufe TI99/4A mir orig. Box TI-Drive, 2 Slim-Line-Drives, 32-KRam, RS232, TEX-Forth (D), TI-Writer, EX-Basic, APESOFT-Grafic, E/A, Joysticks, div. Software u. Bücher auch einzeln!
Liste bei: R. Haberzttl, Goethstr. 20, 85 Nürnberg 10

Verkaufe Ext. RS232 von Atronic mit 32K, Funkschau ab 1963, Elo ab Heft 1 und Chip.
Suche Modulexpander und Exbasic 2 Plus
Tel. 04331/6471 ab 18.00

Verkaufe über 400 DIN A4 Seiten deutsche Beschreibung für den Editor Assembler mit Tabellen und Beispielen
A. Pack, Eickelerstr. 60, 4690 Herne 2

Billig zu verkaufen!
Konsul+Recorder+Anschluß-Kabel 150,–
Konsol + Extended Basic + Recorder 300,–
Recorder Extra 60,–
Extended Basic extra!! 175,–
Floppy (nur Laufwerk)!! BASF 200,–
Schach+Demolition Division 55,–
Datenverwaltung + Alien Addition 60,–
Statistik + Minus Mission 50,–
Beginning Grammar + Alligator MIX 40,–
Parsec + Demolition Division 40,–
Donkey Kong (von Atari) 20,–
Yahzee (Kniffel) 20,–
Tombstone City 20,–
Pack Man 25,–
Tip Trick (Buch) 25,–
Buch für Anfänger und Fortgeschrittene 25,–
TI REVUE seit ab Erste 20,–
Versandliste 50,–
Rechnungsstellung 50,–
Lagerverwaltung 50,–
Joystick (Original Texas einpaar) 50,–
Orhan Kiran

Verkaufe TI 99/4 A, 2 1/2 Jahre alt, kaum benutzt + Cassrec. + 7 Cassetten mit Spielen + Lernprogrammen + viel Literatur + Fachzeitschriften + Massig Listings + 2 Atari Joysticks + 2 Module (Connect Four + Ghrisholm Trail) mit sämtlichen Anschlußkabeln für 250,– DM Tel. 02733-3571

TI 99/4A: Ext. Doppellaufwerk (Pertec) in Box + ext. Discontr. (Texasi) + Diskmanager 880,–; RS232 (Atronic) ext. 2 Ports 240,– ext. 32K + Centronic. (Atronic) 220,–. Tel. 06182/26186

Orig. USA ExtB + Dt Handb 150,–
Orig. Datenverw. u. Analyse 50,–
Orig. TI-Joysticks 25,–
Chip-Spezial Listings 10,–
TI-Spezial Sonderh. 1 6,–
TI-Revue 2/84–1/86 komplett 25,–
Tel.: 040/5238242 Stowasser

Ich suche im Raum Dortmund Kontakt mit anderen TI 44-Besitzern zum Programme und Probleme Austauschen. Habe Monitorkabel zu verkaufen. Andreas Berkenbroek, Evinger Berg 8, 4600 Dortmund 16

BÖRSE

Suche orig. TI Joysticks bis 25,– DM Tel. 02592/61816

Suche Tauschpartner in TI Basic und verkaufe Spielmodule: Alpiner und Tene Attack für je 30 DM, Tel. 02592/61816

Verk. TI99/4A-Konsole + Expansion System + Laufw. + Disk-cont. + Ex Basic-Disk Manager 2 + 2 Joysticker + Module, Schachmeister + Othello + TI-Revue + Bücher für 1200,– DM. Tel. 0431/726488

Original TI-Module „Datenverwaltung" & „Statistik" mit Anleitungen, sowie TI-Buch „Basic/Extended Basic" zusammen mit DM 100,– abzugeben.
Emde, Schulstr. 3, 5810 Witten, Tel. 02302/79353

Suche! Beschreibung/Anleitung für Wycove Forth, 3D-World Graphicprogr. und Exbasic 2 & Painter (Kull Software auf DSK) in deutsch. Angebote Tel. 0208-70748.
Verkaufe TI Basic/Ex Basic Buch orig. TI in deutsch. Schuller, Mühlenstr. 170, 4330 Mülheim 1

Als Besitzer von TI 99/4A mit Erw.box, Disk.-laufwerken und angeschl. Schreibmaschine suche ich Kontakt zu einem TI Kenner und Könner m. gleichartiger Ausrüstung im Raum Mannheim. Ich erbitte ausschließlich schriftliche Kontaktaufnahme, da ständig unterwegs. Jürgen Müller-Ibold, Lucas-Cranach-Str. 3, 6909 Walldorf/Baden

---

Verk. TI99/4A Rec-Kabel + Abdeckhaube + Literatur + Software Preis VB superbillig. Jörg Storm, Christleseeweg 34, 89 Augsburg 0821/63857

Verkaufe Drucker Seikosha GP [illegible] A Mark II + Pio- [illegible]rafikpgm TI-Artist [illegible]mit Hardcopy für [illegible]0A Preis 400 DM VB [illegible]lfgang Bürger, 5620 [illegible]elbert 1, Birther Str. 35, Tel. 02051 66950 nach 20 Uhr

---

Endlich! Schnittstelle I/0 für alle Ausbaustufen. Zählen, Messen, Steuern, Regeln mit der Konsole INFO 1,50 in Br.marken
G. Haferkorn, Schwarzenbeker Ring 50, HH 73

+Suche: Intakten Sprachsynthesizer für TI 99/4A und defekte Module und Modulgehäuse. Angebot mit Preisangabe an: Peter Rieger, Talstraße 64, 6750 Kaiserslautern

---

**** 32K aceugepuffert *** eingebaut in Konsole oder baue 32K in Ihre Konsole ein Biete: Orig. APESOFT-GRAIFC-Erweit., Tennis, Miner 2049 in Assem.; Flugsim. uvm. auf Disk
Preis VS. Tel.: 069/531595

Verkaufe 300 Programme zu folgenden Preisen:
100 Progr. 50,– DM
200 Progr. 80,– DM
300 Progr. 110,– DM
Liste bei: Schneider Manfred; Nunkircher Str. 42a, 6648 Wadern-Büschfeld

---

Tausche meine Software gegen Module zu meinen Verkaufspreisen. Nähere Information bei Manfred Schneider; Nunkircher Str. 42a; 6648 Wadern-Büschfeld/Saar
Tel. 06874/6705

Suche Hardware im Tausch gegen Software (über 300 Programme stehen Euch zur Auswahl). Näheres bei: Manfred Schneider, Nunkircher Str. 42a, 6648 Wadern-Büschfeld/Saar; Tel. 06874/6705

Verkaufe Alien-Addition zum Preis von nur 10,– DM.
Zu kaufen bei: Manfred Schneider, Nunkircher Str. 42a, 6648 Wadern-Büschfeld, Tel. 06874/6705

Verkaufe TI-99/4a 150,– Allig.Mix/Yahtzee/ZeroZap je 15,–. Amazing/Blasto/CarWars/Hustle NumberMagic/Super DemonAttack TI-Inv. / Vid. Gamesl je 20,– ConnectFour/Parsec/Sneggit je 25,– BuckRodgers/Jungle Hunt/MASH Microsurgeon/Miner2049er Moonsweeper/Munchmobile/Othello Pers. Rec. Keeping/Pers.Rep. Generator Star Trek je 30,–. Ex-Basic (Orig.) 100,– Tandberg 62 ohne Gehäuse 200,–. J. Schneiderat, 0211-632042 (9-15 Uhr werktags)

Schnelle universelle Datenverwaltung mit Kassette. Ex-Basic-DV-Programm auf Kassette gegen 10,– DM von Peter Hielscher, Am Wall 22, 4401 Saerbeck

---

ZAUBERSCHLOSS
Das original Adventure aus der TI-Süecial 4/86 !!!! Kein abgetipptes Programm und somit ohne Fehler.
Direkt vom Autor:
Thorsten Ziehm, An der Au 17, 2081 Haseldorf, Telefon 04129/488

DAS SUPER-ANGEBOT
VERKAUFE: Minimemory, Attack TI-Invaders, Carwars, Othello Pers. Rec. Keep., Pers. Rep. Gen. Minus-Mission, Donkey-Kong. Alle Module mit original Verpakkung und Beschreibung.
Thorsten Ziehm, An der Au 17, 2081 Haseldorf, Tel. 04129/488

DAS SUPER-ANGEBOT
Drucke Ihre Progr. (Disk od. Kassette)
Drucke Adressaufkleber!!
Antilistschuth (Ex-Basic)
Doppelseitige Disk-Nutzung
Thorsten Ziehm, An der Au 17, 2081 Haseldorf, Tel.: 04129/488

Verkaufe TI99/4A (100 DM) + orig. Modulbox (200 DM) + orig. TI-Diskkontroller (intern) (200 DM) + TI-Laufwerk f. Box (350 DM) + 32-K-RAM (intern) 250 DM) + Ext. Basic (100 DM) + Disk-Manager II + Recorderkabel (10 DM) + 6 Handbücher (50 DM). Bei Komplettabnahme: dM 1100,– Tel. 0221/843623

Verk. Modul Video Games 10,– Buch Progr. für den TI 99/4A von R. Heigenmoser 12,– Buch 36 TI 99 Progr. 8,– von Len Turner Sybex Bücher Band 1&2 mti je 100 Progr. von D.H. Ahl a 10,–
Tel. 02355/7607

TI99/4A+Ext-Basic 360 DM (nur zusammen); Module Statistik, Report-Generator, Datenverwaltung je 75 DM, Minus Mission, Alien Addition je 25 DM, Rec-Kabel 15 DM Lernkassetten Basic-ExBasic je 15 DM
Bei Nachnahme plus 2,50 DM
Klaus Lindner, Weilerstr. 7 5303 Bornheim 4, Tel. 02227/7040

ACHTUNG ASSEMBLER-FREAKS! Suche Kontakt zu E/A-Usern zwecks Informations- und Programmaustausch. Weitere Interessengebiete: DFÜ u. Hardwareselbstbau. G. Morgenthaler, Angelstr. 24, 7143 Vaihingen/Enz

---

Achtung!!!
Das Programm Samurai aus der TI-Spezial Nr. 4/86 jetzt vom Autor !!!
Gegen 10 DM im Briefumschlag sende ich es auf Kassette zu! – Super Sound – Super Grafik – zwei Teile – keine Speichererweiterung nötig – nur Ext.Bas. – 22,1 Byte – anfordern bei:
H.H.T.S.S
Christoph Hinsch
Achtern Diek 5
2082 Uetersen!!!
– Bei Anfragen Rückporto beilegen!!!

Verk. XB für 150,–, Speech synthi DM 100,–, Suche XB II Plus. M. Bannert – Mozartstr. 5 – 8676 Schwarzenbach – 09284/8338
Verk. Software (TIB, XB, MC) z.B.: XB-Logo mit 32 Commands!! Joystickadapter nur DM 25,–!! Adv-Modul auf Cass. DM 25,–!! 09284-8338 !!
Suche Spielmodule auf Cass. Ladezeit egal! Erstelle Programme nach Wunsch!
M. Bannert, Mozartstr. 5, 8676 Schwarzenbach, T. 09284/8338

Verkaufe Assembler-Kurs für TI99 auf Cassette mit 2 Handbüchern für 50,– DM. Matthias Orf, Birkenallee 34, D-3507 Braunatal I, Tel. 0561/497990

32k in die Konsole, Druckerkabel, Einbau von Resetknöpfen, Zehnertastaturen, Hardcopyausdrucke, Prg,-Speicher, Clubheft, uvm. gibt es für einen geringen Beitrag im TI-CLUB BAUNATAL, Matthias Orf, Birkenallee 34, D-3507 Baunatal I, Tel. 0561/497990. Info 50 Pfg..oder Clubheft 30,– DM anfordern.

Drucke Ihre Listings für nur 4,– DM je Programm. Auch ausdrucken von Listen aus Datenverwaltung & Analyse, Multiplan, TI-Writer ect. möglich. Cassette doer Disk + Geld an Matthias Orf, Birkenallee 34, D-3507 Baunatal 1, Tel. 0561/497990 ... oder Info 0,80 DM anfordern.

Verk. RS232/Drucker Kabel/TI-Forth + Buch/Buch TI intern/Daten Recorder/TI Adv. Spiele/Ti-Rechnungst./Zero-Zap/Tausche Progr./Div. Magazine TI-Soft Club Belgien/Suche Beschr. zu ID-Data/D-Conto/Verk. TI Cassetten Basickurs/07141/461147